ENABLING METHODS FOR THE DESIGN AND OPTIMIZATION OF DETECTION ARCHITECTURES

A Dissertation Thesis
Presented to
The Academic Faculty

By

Alexia Payan

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in Aerospace Engineering

Georgia Institute of Technology

May, 2013

ENABLING METHODS FOR THE DESIGN AND OPTIMIZATION OF DETECTION ARCHITECTURES

Approved by:

Prof. Dimitri N. Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Prof. Vitali Volovoi
School of Aerospace Engineering
Georgia Institute of Technology

Prof. Elena Garcia
School of Aerospace Engineering
Georgia Institute of Technology

LTC U.S. Army Chuck Stancil (Ret.)
Georgia Tech Research Institute
Georgia Institute of Technology

Prof. Brian German
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: March 29th, 2013

AKNOWLEDGEMENTS

As I am about to conclude this journey towards my Ph.D., I would like to take the time to thank those people who have made it possible for me to be where I am and who I have become.

First, I would like to express my profound gratitude to my research advisor, Dr. Dimitri Mavris for his continued guidance and his enormous heart. His kindness has always been greatly appreciated and cherished. Merci, Doc! I have learned so much with you and from you. I thank you for your dedication to making the personal development of your students a priority.

Second, I would like to express my kind regards to Dr. Elena Garcia for her time, her help, her gentle words and smiles as she supervised my research throughout my journey at ASDL. Elena, thank you for your enthusiasm, your dedication, and your positive attitude.

Third, I would like to thank the other members of my committee, Dr. German, Dr. Volovoi, and Lieutenant Colonel Chuck Stancil for their time, their advice, and their help in making this research a better piece of work.

I would also like to thank the Thales Group and EADS North America for financially supporting distinctive parts of this research, as well as Romain Lamour and Arnaud Thioulouse for their help.

Finally, I would like to acknowledge all my friends at ASDL and at Georgia Tech that have brightened this journey and made it more enjoyable.

Last but not least, to my family, I would like to say thank you from the bottom of my heart for believing in me, for your love and your reassuring support throughout all this years spent away from all of you; to my Babybou, I thank you for your love, your help, your comforting presence, your playful attitude, and your kindness throughout these years of hard labor.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

ABM	Agent-Based Modeling
ASDL	Aerospace Systems Design Laboratory
CAP	Critical Assets Protection
CIP	Critical Infrastructures Protection
CBP	Customs and Border Protection
DGA	Délégation Générale pour l'Armement (French General Delegation for the Armament)
DHS	Department of Homeland Security
DoE	Design of Experiments
DoD	Department of Defense
DODA	Design and Optimization of Detection Architectures
ECPD	Engineers' Council for Professional Development
FEMA	Federal Emergency Management Agency
GA	Genetic Algorithm
HCC	High Cost Camera
HCR	High Cost Radar
HLCCM	High-Level Cross-Consistency Matrix
HLMM	High-Level Morphological Matrix
HS	Homeland Security
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council on Systems Engineering
LCC	Low Cost Camera
LCR	Low Cost Radar
MCC	Medium Cost Camera

MCR	Medium Cost Radar
PSO	Particle Swarm Optimization
M&S	Modeling and Simulation
MS&O	Modeling, Simulation, and Optimization
NATO	North Atlantic Treaty Organization
RCS	Radar Cross Section
SA	Simulated Annealing
SE	Systems Engineering
SEP	Systems Engineering Process
SLCCM	Sub-Level Cross-Consistency Matrix
SLMM	Sub-Level Morphological Matrix
S_MAX	Maximum Number of Systems Allowed in a Detection Architecture
SoS	System-Of-Systems
U.S.	United States of America
USACE	United States Army Corps of Engineers
USDOT	United States Department of Transportation
DSN	Distributed Sensor Network
WSN	Wireless Sensor Network

SUMMARY

The surveillance of geographic borders and critical infrastructures using limited sensor capability has always been a challenging task in many homeland security applications. Geographic borders may be very long and may go through isolated areas that are sometimes uninhabitated. Critical assets may be large and numerous and may be located in highly populated areas. As a result, it is virtually impossible to secure both each and every mile of border around the country, and each and every critical infrastructure inside the country. Most often, a compromise must be made between the percentage of asset covered by surveillance systems and the induced cost. However, various entities may have distinct performance and cost expectations for the surveillance architecture. This difference in preference needs to be captured in order to determine surveillance and detection architectures able to provide the required protection at the projected level of cost. Although threats to homeland security can be conceived to take place in many forms, those regarding illegal penetration of the homeland under the cover of day-to-day activities have been identified to be of particular interest by the U.S. and several European governments. For instance, the proliferation of low altitude aerial systems, combined with regular air traffic growth, poses a unique challenge for the surveillance of homeland airspace and in particular for identifying potentially hostile vehicles interoperating with friendly aircraft. Similarly, the proliferation of drug smuggling, illegal immigration, international organized crime, and more recently, modern piracy, require the strengthening of land border and maritime awareness, and point to increasingly complex and challenging national and coastal security environments. In this

context, it is critical to be able to monitor, collect information (i.e. detect, identify) and eventually intercept suspicious entities or systems well before they reach the border or strategic land and coastal sites. Nevertheless, suspicious aerial, ground or maritime systems can efficiently interoperate with friendly systems so as to compromise the situational awareness of border and maritime protection missions. As a consequence, it is necessary to comprehensively understand what composes the aerial, ground and maritime domains to determine a strategy of action.

The complexity and challenges associated with the above mission and with the protection of the homeland may explain why a methodology enabling the design, modeling, simulation, and optimization of detection architectures able to provide accurate scanning of the air, land, and maritime domains, in specific geographic and climatic environments, is a capital concern for the defense and protection community. As a result, the present thesis focuses on the development of surveillance architectures of distributed ground platforms and sensors for detecting aerial, land, and maritime systems. This task primarily involves identifying the best combination and positioning of detection and surveillance systems able to monitor the homeland and its shores, and to collect information about the surrounding aerial, terrestrial and maritime environments. To do so, it is imperative to quantitatively assess current state-of-the-art as well as future notional systems, generate meaningful comparisons across disparate platforms, explore tradeoffs between a myriad of factors, and identify key technological gaps.

This thesis proposes a seven-step methodology aimed at addressing the aforementioned gaps and challenges. The methodology facilitates the traceable, structured, and reproducible design, Modeling, Simulation, and Optimization of

Distributed Detection system architectures for surveillance missions in the context of homeland security, and is thus named M-SODDA. M-SODDA considers a set of heterogeneous detection systems and distributes them over large areas of interest in order to design detection architectures providing the maximum global detection coverage at reasonable costs in specific topographic and climatic environments. Additionally, it enables the decision maker to really understand the nature of the detection architectures, assess their capabilities through a number of notional "what-if" scenarios and analyze the relative sensitivity of trade-offs at the architecture level, the system level, and the operational level.

More precisely, M-SODDA reformulates the homeland security problem in clear terms so as to facilitate the subsequent modeling and simulation of potential operational scenarios of interest. The problem is first decomposed into its main elements and relevant alternatives and attributes capturing the whole spectrum of possibilities are investigated. Then, the operational compatibilities between pairs of elements, alternatives and attributes are assessed to create notional, yet realistic, simulation scenarios. Next, a detailed description of a multidisciplinary strategy for the design and optimization of distributed detection system architectures in terms of detection performance and cost is provided. This implies the creation of a framework for the modeling and simulation of previously selected scenarios, as well as the development of improved methods for the rapid optimization of detection architectures in specific operational situations. In other words, the present thesis describes a new approach to determining detection architectures able to provide effective coverage of a given geographical environment at a minimum cost. This is done by optimizing the composition and the geometrical structure of the detection system-of-systems. To do so, two candidate evolutionary optimization approaches (genetic algorithm and particle swarm optimization) are selected to potentially solve the homeland security optimization problem of interest. The optimization algorithms are then adapted and refined from their original versions in order to account for the peculiar characteristics of the homeland security application. Next, appropriate sets of algorithm parameters adapted to the homeland security mission are obtained by applying the modified optimization algorithms to simpler analytical test problems (whose solutions are known) presenting similar properties as the original problem. It is found that the resulting set of algorithm parameter values, able to ensure the convergence of the optimization algorithms to accurate distributed system architecture solutions is dictated by the maximum size of the detection architecture. The convergence properties of the candidate evolutionary optimization approaches are further studied and compared to determine the optimization method which globally presents the best performance and the lowest computational cost. It is found that the particle swarm optimization algorithm is better suited to find solutions to the original optimization problem. Subsequently, a heuristic recursive optimization scheme is developed to check the accuracy of the solutions provided by the modified particle swarm optimization algorithm when applied to the original homeland security application. It is found that the modified algorithm is successful at finding reliable detection architectures able to satisfy the constraints of the homeland security mission scenario. In other words, the adapted particle swarm optimization method is able to efficiently optimize concurrently the number, the types, the properties, and the positions of a set of heterogeneous detection systems over a large area of operations for a specific homeland security mission, given performance requirements and/or cost constraints. At that point, the detection architecture solutions are composed of sensor systems having a fixed position on the theater of operations. In a final step, the modeling and simulation framework is used to rapidly, quantitatively, and efficiently evaluate the operational effectiveness of a portfolio of fixed Pareto efficient detection architectures obtained with the modified particle swarm optimization algorithm. It is found that complementing the fixed detection architectures with mobile detection systems transported by patrol units notably enhances their coverage and operational performance. Finally, the modeling and simulation framework is used to obtain a Pareto frontier of distributed detection system architectures able to satisfy varying customer preferences on coverage performance and related cost.

CHAPTER I

MOTIVATION

1.1. Introduction

The word engineering comes from the Latin *ingenerare* which means "to create." Engineering is the discipline of acquiring and applying scientific and technical knowledge to the design, analysis, and/or construction of works for practical purposes. It is the application of science to the optimum conversion of the resources of nature to the uses of humankind. The American Engineers' Council for Professional Development, also known as ECPD, defines Engineering as: *"The creative application of scientific principles to design or develop structures, machines, apparatus, or manufacturing processes, or works utilizing them singly or in combination; or to construct or operate the same with full cognizance of their design; or to forecast their behavior under specific operating conditions; all as respects an intended function, economics of operation and safety to life and property."* [0, [2], [3]].

As such, while the function of the scientist is to know, that of the engineer is to do. Unlike the scientist, the engineer is not free to select the problem that interests him; he must solve problems as they arise; his solution must satisfy conflicting requirements. Usually efficiency costs money; safety adds to complexity; improved performance increases weight. The engineering solution should hopefully be the most robust solution, and always be the end result that, taking many controllable and uncontrollable factors into account, is most desirable.

However, it is not always easy to determine the most robust solution to a problem especially when it comes to problems involving complex, interacting or disparate systems. Indeed, engineering problems span domains and timescales, with complexity that changes dramatically with the context of evolution of the world. As engineering knowledge and experience advance, man has been able to better design systems meeting his needs, in fields more challenging each time. As systems become more and more complex, the focus is now on designing a robust system-of-systems composed of elements that may be different in their nature, but almost always interacting and interoperating. These elements need to be associated in a particular manner to reach a desired outcome.

Such a challenge dominates thinking in the defense community of western countries: How can heterogeneous systems best be distributed over large areas to provide adequate global coverage at a reasonable cost in the context of homeland security?

Indeed, the risks associated with interacting with foreign entities, whether it be by air, by land, or by sea keeps increasing and people struggle to find a way to protect borders, civil vital assets, population areas, land and coastal sites, as well as strategic sites and critical infrastructures. The sharing of information and the rapid evolution of technology not only enhance our life, but also confound hostile systems with friendly systems, thus increasing the lethality of threats to the national security of first world nations. Moreover, the development of sophisticated and advanced technologies and methods, and their availability to individuals engaged in criminal activities, pose an extreme challenge to the detection of potential malevolent entities hiding among daily activities and systems.

Therefore, enhancing the capabilities of detection of items of interest (IoIs) that could be a threat to the populations and the nations' assets is a necessity. This involves identifying the best positioning of detection systems enabling the protection of land and maritime borders with neighboring countries, as well as key assets and populations, in a timely fashion. However, this is hindered by the complex nature of the system-of-systems required to provide an adequate protection. Therefore, current systems used for the protection of borders, of populations and of national assets are generally not adapted to and not optimized for the problem at hand. Indeed, their efficient combination is usually hindered by the size of the assets to protect and by the cost required to protect them to an acceptable level. Besides, the assets requiring protection against potential malevolent systems are highly heterogeneous in nature, as are the potential threats to these assets.

That may explain why as new threats and risks continue to emerge, armed forces are being called upon to secure and protect people and assets anywhere in the world. Such protection is no longer limited to times of war; civil unrest, peacekeeping, border protection, and anti-terrorism operations are the new focus of developed nations. Thus, increasingly rapid and flexible deployment of interrelating and interoperating systems, beyond normal theaters of operation to extremely diverse urban and rural environments, is required. This is why the definition of a portfolio of sensor systems able to provide adequate protection in a specific context is of high interest and is investigated as part of this work.

1.2. Problem Definition and Motivation

"The world will not be destroyed by those who do evil, but by those who watch and do nothing."

- Albert Einstein

During the World Wars, systems were designed to counter known adversaries with predictable doctrines and strategies. The current strategic environment, on the other hand, is dominated by uncertainty and driven by opponents that seek to exploit non-traditional weaknesses in the national security of developed countries.

The first step in providing an efficient protection of the nations' borders, key assets and populations, is to better delineate the scope of the problem and try to understand each term involved.

1.2.1. Homeland Security

"[NATO] An Alliance in which Europe and North America are consulting every day on the key security issues before them. Acting together, in the field, to defend our shared security… Because in a dangerous world, business as usual is not an option."

- NATO Secretary General, Jaap de Hoop Scheffer
NDA Conference, May 17, 2004

First and foremost, the general context of this study is homeland security. Homeland security *"refers to domestic government actions against the threats of terrorism. This term became prominent following the 9/11 attack. It includes emergency mobilization, including volunteer medical, police, and fire personnel. It also includes new domestic surveillance efforts, secret arrests and detention, and infrastructure protection, as well as border control"* [4].

The tenets of homeland security are fundamentally different from those historically defined for national security. Indeed, historically, securing the nation entailed the protection of the armed forces outside the borders of the country. The goal was to "keep the neighborhood safe" in the global and geopolitical sense. Nevertheless, the emergence of international terrorism within the nation's own borders has moved the front line of domestic security to the streets. Faced with the realities and the cruelty of the September 11th attacks in the United States, the mission to protect our homeland now entails "keeping our neighborhood safe" in the most literal sense of the words [5].

Homeland security is further defined by Richard Falkenrath, former Deputy Assistant to the President and Deputy Homeland Security Advisor, as "*a concerted comprehensive and nationwide effort to prevent future terrorist attacks, to protect the most vulnerable targets against future terrorist attacks and to be ready to respond against possible attacks* *and minimize loss of life and damage if such attacks occur*" (NDA Conference, November 17, 2003) [6].

Finally, the "National Strategy for Homeland Security" [4] aligns homeland security efforts into six critical mission areas:

- Intelligence and warning
- Border and transportation security
- Domestic counterterrorism
- Protecting critical infrastructures and key assets
- Defending against catastrophic terrorism
- Emergency preparedness and response

As a summary, homeland security is essentially oriented towards the protection of our Nation's main assets and our populations against potentially harmful activities that could operate by air, by land, or by sea. The main three aspects of homeland security studied in this research concern critical infrastructures protection, key assets monitoring, and border surveillance. These three areas of study are described in more details in the following sections.

1.2.2. Critical Infrastructures

"Critical infrastructures consist of those physical and information technology facilities, networks, services and assets which, if disrupted or destroyed, would have a serious impact on the health, safety, security or economic well-being of citizens or the effective functioning of governments in the member states. Critical infrastructures extend across many sectors of the economy, including banking and finance, transport and distribution, energy, utilities, health, food supply and communications, as well as key government services" [7].

Critical infrastructures are further defined as being those *"systems and assets, whether physical or virtual, so vital to […] [a country] that the incapacity or destruction of such systems and assets would have a debilitating impact on security, national economic security, national public health or safety, or any combination of those matters. More specifically, critical infrastructures are those people, things, or systems that must be intact and operational in order to make daily living and working possible"* [8].

Critical infrastructures can be found in various sectors of activity, each important as regards to the general functioning of a human society. Examples include:

- Energy
- Water Services
- Agriculture and Food
- Communications and Information Technologies
- Public Health Care
- Emergency Services
- Banking and Finance
- Public and Legal Order and Safety
- Civil Administration
- Postal and Shipping
- Transportation
- Chemical and Nuclear Industry and Hazardous Materials and Biological Industries
- Production, Storage and Transport of Dangerous Goods
- Space and Research
- Government
- Defense Industrial Base

Hence, it is vital for a community which depends on critical infrastructures for economic, security, quality of life, delivery of service and governance to protect such vital assets.

Indeed, the Homeland Security Presidential Directive 7 [9] clearly states that *"Critical infrastructure and key resources provide the essential services that underpin [...] [the] society. The Nation possesses numerous key resources, whose exploitation or destruction by terrorists could cause catastrophic health effects or mass casualties comparable to those from the use of a weapon of mass destruction, or could profoundly affect our national prestige and morale. In addition, there is critical infrastructure so vital that its incapacitation, exploitation, or destruction, through terrorist attack, could have a debilitating effect on security and economic well-being."*

1.2.2.1. The Case for Critical Infrastructures

Critical infrastructures frame our daily lives and enable us to enjoy one of the highest overall standards of life in the world. Needless to say that when we flip a switch, we expect light; when we pick up the receiver, we expect a dial tone; when we turn on a tap, we expect drinkable water; when we open the refrigerator, we expect to find fresh food; when we go to the pharmacy, we expect to find medicines. Electricity, telecommunications, clean water, food, medicines are only a few of the critical infrastructure services that we tend to take for granted. They are so well ingrained in our daily life that it is only when service is disrupted that we notice them. In such cases, we expect reasonable explanations and rapid restoration of service.

As such, critical infrastructure sectors provide goods and services that contribute to a strong national defense and thriving economy. They also create a sense of confidence in the society, and their reliability, robustness and resiliency form a major part of the national identity and strategic purpose.

However, whether it be in the United States or in Europe, critical infrastructures are highly sophisticated, complex, interdependent and connected. They consist of human capital and physical and cyber systems working together in processes that are highly interdependent. This is the result of the concentration of populations in urban areas, the rationalization of the industry, the consolidation of corporation, and the adoption of efficient business practices, especially in Europe.

To complicate the matter further, critical infrastructure components are often dispersed over large areas and are typically interconnected. They represent a true system-of-systems as they dependent on the continued availability and operation of other dynamic systems and functions. Critical infrastructures have become more dependent on common information technologies such as mobile phones, internet and space-based radio-navigation and communication. Hence, it is not difficult to conceive that problems can cascade through these interdependent infrastructures, causing unexpected and increasingly more serious failures of essential services. The interconnectedness and interdependence existing between critical infrastructures, along with their sheer numbers and pervasiveness, make them more vulnerable to disruption or destruction. The dynamic nature of these interdependent infrastructures and the extent to which our daily life relies on them, make a terrorist attack to destroy or disrupt them have a tremendous impact beyond the immediate target: the effects of the attack could indeed continue to reverberate long after the immediate damage is done [7], [10]. Nevertheless, given the size and scope of the potential target set for terrorists, one cannot assume that it will be possible to protect them all, completely, at all times, against all conceivable types of threats. Even with continuous monitoring and surveillance, threat interdiction remains an elusive and unrealistic goal.

Moreover, trying to identify all possible exposure, to all possible threat events, across all critical components, and forecasting all possible consequences is an unthinkable task. Indeed, while one protective measure is developed against a particular terrorist tactic on a particular target, terrorists are already focusing on the development of a new tactic to attack another target. Hence, in order to be efficient and effective, one needs to have a thorough understanding of these complexities.

1.2.2.2. The Meaning of Criticality

The Commission of the European Communities of 2006 [7] defines three major criteria for identifying critical infrastructures: scope and magnitude of the impact of the loss, and time scale of the impact. Each of these criteria may be defined as follows.

First, the **scope of the impact of the loss of a critical element** is rated by the extent of the geographic area which could be affected by its loss or unavailability, and is separated into four categories: International, National, Provincial/territorial, and Local.

Then, the **magnitude of the impact of the loss of a critical element** is assessed using a qualitative scale, as follows: None, Minimal, Moderate, and Major.

Moreover, some criteria are defined in order to **assess potential magnitude**:

- *Public impact*, namely the amount of population affected, the number of casualties, medical illness, serious injuries, the need of evacuation of a populated area
- *Economic*, namely the effect on the GPD, the significance of economic loss and/or degradation of products or services
- *Environment*, namely the impact on the public and surrounding location
- *Interdependency*, namely the connection of the critical element under consideration with other critical infrastructure elements
- *Political*, namely the confidence in the ability of government to tackle the problem

Finally, the **effects of time** which ascertain at what point in time the loss of a critical element could have a serious impact, are measured using a quantitative scale such as: Immediate, 24-48 hours, One week, One month, One year, and Other.

Taking the example of European countries, all infrastructures vital to the maintenance of primary social and economic processes are considered critical sectors. These **critical sectors** are the following:

- Banking and Finance
- Chemical and Biotechnological Industries

- Energy and Electricity
- Nuclear Power Stations
- Public Health
- Public Safety and Order
- Telecommunications
- Transportation Systems
- Water Supply

In "Risk Management and CIP: Assessing, Integrating, and Managing Threats, Vulnerabilities and Consequences" [11], criticality is typically defined as a measure of the consequences associated with the loss or degradation of a particular asset. Consequences can be categorized as follows: Economic, Financial, Environmental, Health and Safety, Technological, Operational, and Time to recover from an attack.

While the immediate impact is important, the time and the resources required to recover from an attack or, if possible, to replace the lost capability, are quite significant as well. If the loss of an asset results in a large immediate disruption but the asset can be replaced or the service restored quickly and cheaply, or there are cost effective substitutes to the asset and/or the service provided, then the total consequence may not be so great. However, if the loss of an asset results in a small immediate disturbance but that disturbance continues for a long period of time because the capability cannot be reconstituted quickly or is lost forever, then the consequences are significantly more dramatic.

Another issue is the interdependency among infrastructures: the higher the interdependency or dependency of one infrastructure on (an)other(s), the higher the risk of cascading consequences that might affect more than one facility/company/sector. For example, the loss of electric power can cause problems in the water purification process.

Moreover, when assessing the criticality of an infrastructure, it is necessary to not only look at the dependent assets but also to look at the asset(s) on which the considered infrastructure depend(s). Indeed, a firm may rely on the output from a specific asset operated by someone else. The user may thus consider this asset critical from his perspective, but the owner and operator of the asset in question may not. Hence, it is necessary to account for the vulnerability of those assets owned or operated by someone else that provide critical input(s) into (an)other system(s).

Consequently, the interdependency issue can be seen as both technical and political or legal in the sense that it might be possible to induce an entity to protect another entity not considered critical by its owner or operator but critical for the former entity.

1.2.3. Key Assets

Key assets and high profile events are also potential targets for terrorist attacks that, in the worst case scenarios, could result in not only loss of a large number of lives and in property destruction, but also in profound damage to the nation's prestige, morale and confidence. Even if it appears that key assets such as nuclear power plants, dams and hazardous materials storage facilities are not vital to the continuity of critical services within the nation, a successful strike on such targets could affect public health and highly damage public safety in the long term, in addition to human casualties and property destruction.

Other key assets are simply part of the major symbols of the nation: national icons, monuments, museums, cultural buildings and historical attractions preserve history and culture, honor achievements, and represent the natural grandeur of a country. They celebrate the nation's ideals and way of life, and present attractive targets to terrorists since they are highly coupled with national events and celebratory activities that bring together a significant amount of people at the same place and time.

Another category of key assets includes facilities and structures that represent national economic power and technological advancement such as hazardous materials, fuels and chemicals storage facilities. Destruction or disruption of such facilities could have a significant impact on the public health and safety, as well as on national confidence and on the economy.

A last category of key assets includes those places where large numbers of people regularly gather to conduct business activities or personal transactions, to shop or to enjoy themselves, such as commercial centers, sport stadiums, office buildings, national parks and amusement parks. Given the national-level interest for these sites and the potential consequences on human life and societal well being, it is imperative to protect them to prevent fatalities and to preserve public confidence.

In short, key assets represent such a broad array of unique facilities, sites and structures that, if they happen to be destroyed or disrupted, it could do great harm in multiple dimensions [5].

1.2.4. National Borders

The national land and maritime borders are being secured by the National Border Patrol under the U.S. Customs and Border Protection (CBP) agency [12]. CBP is a special component of the Department of Homeland Security (DHS) tasked at deterring, detecting, and preventing threats from entering the country while facilitating legitimate trade and travel. This involves countering criminal and terrorist activities such as the illegal "exploitation of international passenger and commercial cargo transportation systems at 327 official air, land, and sea (POEs)." [12] CBP is also responsible for securing about 7,000 miles of land border and 95,000 miles of shoreline in collaboration with the U.S. Coast Guard. While its primary mission is the prevention of terrorism, the U.S. CBP is also responsible for apprehending illegal immigrants, drug and human smugglers, and other contraband-related individuals to protect the intellectual, economic and agricultural interests of the country [13].

In this context, the National Border Patrol's strategy consists of five main objectives, described as follows [12].

- *Preventing* terrorists and terrorist weapons from entering the United States by detecting and apprehending terrorists and their weapons as they attempt to cross land and/or maritime borders. This may be done by enhancing partnership with other federal, state, local, and tribal law enforcement agencies, as well strategically deploying sensor systems able to detect, respond and interdict illegal border crossings.

- *Strengthening, maintaining and expanding* the operational control of borders to deter and prevent illegal entries of terrorists, terrorist weapons, drug smugglers, and illegal aliens into the United States. This may be done by the strategic deployment of border patrol personnel, equipment, technology, and infrastructure items such as roads, lights and fences. This has already proven to be an efficient approach in high-traffic corridors and high-threat areas along the southwest border with Mexico where illegal border crossings and smuggling is a serious issue.
- *Protecting* the citizens by prohibiting the introduction of illegal drugs, harmful materials and organisms, and by deterring human smuggling and other contraband activities. This may be done by deploying adequate resources in concerned geographic environments so as to detect, interdict, and respond to potential smuggling threats. This entails collecting, sharing, and processing related information, data and intelligence, in partnership with other centers and agencies such as the National Targeting Center (NTC) and the Immigration and Customs Enforcement (ICE) agency.
- *Making use* of "Smart Border" technology to enhance situational awareness, monitor, detect, respond to, and identify potential threats. This implies an appropriate coordination of activities between border patrol personnel and technology such as camera systems for day/night and infrared detection, radar platforms, aerial platforms, and other portable detection devices.
- *Reducing* crime in border communities and improving the quality of life and economic well-being of concerned areas. This involves not only deploying resources in areas deemed high-threat or high-priority, but also conducting public outreach to educate populations about the risks associated with helping terrorists and smugglers to enter the country.

The primary challenges faced by the U.S. Customs and Border Patrol consist in addressing dynamic threats and vulnerabilities by improving detection and surveillance capabilities able to complement and support existing agent resources, and by increasing the cooperation with Mexico and Canada to improve safety and slow illegal entries.

1.2.4.1. National Security

In terms of national security, CBP faces the challenge of maintaining its vigilance in screening international travelers in order to try to uncover terrorist groups or individuals willing to harm the country. Such malevolent entities are indeed constantly improving their techniques to inflict harm on the country by cooperating with local terrorist groups, by intensifying their efforts to operate directly from the country they are targeting, and by attempting to acquire chemical, biological, radiological, and nuclear material to build weapons of mass destruction. Therefore, the CBP must adapt its plans and operations to detect and interdict terrorist threats to the public safety and security. In doing so, the CBP must adopt a multilayered approach in which it receives a-priori information on cargo, people, and other commercial goods about to enter the country, it identifies high-risk shipments, individuals, and other commercial goods with automated targeting systems and advanced inspection technologies, and it extends its authority outside of the country's borders by cooperating with foreign law enforcement entities. Securing the country's borders thus demands a complex, risk-based, layered approach. Gaining and maintaining control of the borders requires the deployment of a robust mix of technology, law enforcement personnel, and tactical infrastructure, as well intelligence and strong partnership with federal, state, local, tribal, and foreign governments [14].

Moreover, illegal immigration and smuggling compromises the national security by creating pathways for illegal entry, demand for false documentation and identities. While most illegal immigrants and drug smugglers do not pose a direct threat to national security, they do provide underground networks for terrorists to successfully blend into the national population [15]. This is why controlling the flow of illegal immigrants across the borders is critical and challenging at the same time, and requires a proper mix of resources to enhance the ability of CBP to detect, apprehend, and prosecute people illegally crossing the borders in high-traffic and high-threat areas.

Illegal drug smuggling and trafficking complicates the problem further in that they flow in both directions across the country's borders and make use of both private and commercial vehicles, low-flying aircraft, small boats and all-terrain vehicles, as well as human transporters. This makes it laborious for law enforcement agencies to distinguish between legitimate activities and suspicious entities despite the current radar technology deployed along the borders from which people, vehicles, and low-flying aircraft can escape detection in extreme terrains.

1.2.4.2. Regional Threat Profile

In this fight for national safety and security, the border is not merely physical. Maintaining control of and securing the borders thus involves paying attention to each specific region of the borders and to what happens far outside the country. Border security is a continuum of actions relying on the geographical border as the last line of defense rather than the first. Each region of the geographical border requires focused analysis, strategies and implementation plans that are tailored to address problems specific to that region [13]. For instance, three main regions can be indentified in the U.S. border: the northern border region with Canada, the southwest border region with Mexico, and the southeast coastal border region with the Caribbean countries and the Central American countries through the Gulf of Mexico [12].

- The northern border region corresponds to the area running between Canada and the United States from Washington state through the Great Lakes Region to the state of Maine. Although the border is neither actively patrolled nor militarized, illegal immigration and smuggling regularly occur from major Canadian cities that are proximate to the U.S. border. Furthermore, known extremist and terrorist groups have been historically present along the northern border both in Canada and in the United States.
- The southwest border region spans more than 2,000 miles between the United States and Mexico. The terrain around the border is typically extremely harsh and inhospitable which represents a challenge to border security missions. The southwest border region provides relatively easy access into the United States for the most common transnational threats such as drug trafficking, human smuggling, and terrorism. The southwest border is currently equipped with 33 legitimate ports of entry separated by hundreds of miles of open desert,

mountains, the Rio Grande River, and the corresponding coastal waters. These scarcely populated areas provide a perfect environment for cross-border illegal and criminal activities. The most rampant national security vulnerability concerns drug and human trafficking which exploits the border in both directions: from Mexico to the United States to smuggle drug and people, and from the United States to Mexico to move money and weapons.

- The southeast coastal region which extends for more than 2,000 miles, represents a unique detection and surveillance challenge for ground, aircraft, and sea patrols due to the difficulty at effectively maintaining a comprehensive situational awareness of low-flying aircraft and water-surface vehicles over large geographic areas. The major threat faced by the CBP at the southeast border mainly concerns contraband smuggling originating from the Yucatan peninsula and the Caribbean islands, proceeding to the southern islands of the Bahamas. Aerial and maritime vessels combine their efforts to then transport the merchandise to Florida's western coast and to the rest of the United States. This presents a troublesome threat in that smugglers usually operate at night via aerial and marine vehicles moving at maximum speeds, only stopping to refuel and check for surveillance systems.

1.2.4.3. <u>Customs and Border Protection Goals and Objectives</u>

The first strategic goal of the CBP is to "secure the Nation's borders to protect [the country] from the entry of dangerous people and goods and prevent unlawful trade and travel." [14] The corresponding main objectives are:

- To gain and maintain control of aerial, ground, and marine borders by using a robust mix of technology, people, and tactical infrastructure. Effective control of segments of the border implies being able to detect, identify, classify, satisfactorily respond and resolve suspicious activities and illegal entries into the country. In order to do so, it is necessary to enforce laws and regulations related to immigration, customs, trade, and agriculture. This involves developing a layered defense approach through which robust technology solutions can support and complement personnel and tactical infrastructures such as roads, lights, pedestrian and vehicles fences in their security efforts.

- To detect and interdict the illegal entry of materials, goods, and weapons into the United States through the use of advanced scanning and inspection techniques at ports, airports, seaports, permanent land ports of entry, and international areas where CBP operates. The success of this approach resides in the continuous improvement and deployment of non-intrusive inspection technologies able to detect and interdict weapons, drugs, currency, and other contraband concealed in large commercial containers. This ensures that the CBP can respond to increases in both the local and national threat level as well as in the volume of cargo and goods crossing the borders.
- To prevent the entry of malevolent entities into the United States through the adequate gathering of intelligence, biographical information, and biometric data at ports, airports, seaports, permanent land ports of entry, and international areas where CBP operates. This objective is enabled by intelligence technologies such as the U.S. Visitor and Immigrant Status Indicator Technology (US-VISIT) program and the Automated Biometric Identification System/Integrated Automated Fingerprint Identification Systems (IDENT/IAFIS) which collects and maintains a database of fingerprints and photographs of every one individual entering the United States. This provides an unalterable and unassailable way of preventing identity fraud and identifying people with criminal histories or related to terrorist groups or activities.
- To provide the training and the technological means to the CBP agents on the field to enhance their ability to address crucial interdiction missions. This entails "executing counterterrorism and counternarcotics operations, initiating high-risk arrests, safeguarding Federal assets and personnel", protecting ports of entries, managing high-risk entries, "conducting special interdiction operations", and addressing any other similar tactical events.

The second strategic goal of the CBP is to "ensure the efficient flow of legitimate trade and travel across U.S. borders." [14] The corresponding objectives are:

- To effectively and efficiently process people and goods at ports, airports, and seaports using accurate intelligence and modern inspection and processing techniques. For instance, the Trade Act of 2002, the 24-hour rule, and the SAFE Port Act enable collecting information about passengers and goods arriving in the United States. Then, the Automated Targeting System (ATS) can perform additional analysis on people and shipments requiring closer inspection. As for the Automated Commercial Environment (ACE), it provides the CBP and other law enforcement personnel with intelligence regarding cargo so that is can either be expedited or be further inspected due to potential risk issues. This improves traffic management and efficient movement of passengers and goods at ports, airports, and land ports of entry. Concerning passengers, the Secure Electronic Network for Traveler Rapid Inspection program, developed at the southwest border in collaboration with Mexico, and the NEXUS program, developed in cooperation with Canada, facilitate the process on entry into the United States of low-risk trusted travelers. As for commercial drivers, and commercial air travelers, they are covered from the Free and Secure Trade program and from the Global Entry program respectively. The most recent addition to the above suite of advanced inspection and processing technologies is the coupling of the Enforcement case Tracking System (ENFORCE) with modern biometric techniques. This system utilizes a more accurate fingerprint capture and validation process to expedite clearance at land ports of entry of eligible travelers having no prior disqualifying criminal activity.
- To consistently apply enforcement actions so as to target and deter non compliance with international trade rules. This objective is meant to prevent illicit and unfair trade practices and enterprises while facilitating the movement of goods and people into the country. This is complicated by the ever-increasing demand for international trade and the resulting complex and dynamic commercial environment. To meet this objective, CBP must employ a layered approach involving "state-of-the-art analysis and targeting, international verification,

focused border enforcement, post-entry reviews and audits, and stiff punitive actions." [14]

- To facilitate the effective release of legal commercial cargo by conducting regular reviews and ensuring compliance with trade rules. In order to identify potentially high-risk trade areas and to balance security, risk and efficiency, the CBP prioritizes its efforts on three main principles. The first one concerns trade issues that relate to the economic well-being of the country, and to the public health and safety. The second principle involves trade issues that may present non-compliance, technical problems, lack of automation, and complexity. The third and last principle is meant to provide guidance to commercial traders by formulating regulations, rules, and directives to perform legal trade with the United States.

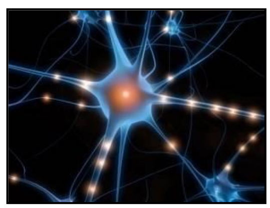
In order to enable the above strategic goals and objectives, it is necessary to put into place three major cross-cutting tools and techniques that will prove critical to CBP's mission success. The first action plan consists in increasing the sharing of intelligence information to maximize border security. This entails collecting real-time data from detection infrastructures at the border as well as from officers and agents on the field so as to help in decision-making. The gathered information can then be reported and disseminated to other critical mission partners and offices. The second action plan requires establishing and strengthening partnerships with federal, state, local, tribal, industrial and international entities to facilitate global trade and travel while complying with agricultural, immigration, and other federal laws and regulations. This is meant to enhance the clearance of cargo, the security of the supply chain, and the compliance with trade rules. To do so, agriculture specialists, Border Patrol agents, CBP officers, and Air and Marine Interdiction agents are being deployed in selected western countries to prevent illegal immigrants, drugs, terrorists, and weapons from entering the country. Their duty is to train foreign government agency personnel to collect information and interdict potentially high-risk people and products from crossing the borders of the United States. On the mainland, CBP cooperates with the U.S. Coast Guard (USCG), the ICE, and other affiliates of the DHS to gain better control of the border by deterring and apprehending suspicious people and products before they are the subject to any criminal activity within the U.S. territory. In addition to the enhancement of partnerships and collaborative efforts to promote security and improve border control, the CBP engages in pursuing new relationships with the U.S. Food and Drug Administration (FDA), the Centers for Disease Control and Prevention (CDCP), and the U.S. Consumer Product Safety Commission (CPSC) in order to protect public health against potential chemical, radiological, nuclear, agricultural, and other products-related threats. The third action plan consists in promoting an environment that leverages “state-of-the-art technologies, innovative strategies, and worldwide partnerships to protect America’s communities and defend its borders.” [14] This involves bringing together several technology-, finance-, resource-, training-, and program-related management practices to create a global framework able to attract a talented and dedicated workforce motivated by an achievement and results-driven culture that fosters integrity. The success of this action plan requires highly trained mission support personnel, supervisors, managers, and executives dedicated to prevention, detection, and investigation. This implies strengthening the measures and processes followed when hiring new personnel by promoting security specialists, polygraph examiners, and behavior analysts so as to enhance and accelerate background checks, deal with corruption and misconduct in the field, and “conduct covert field surveys, inspections, and surveillance in efforts to strengthen integrity.” [14]

Several measures of performance can be brainstormed to capture the effectiveness of the two aforementioned strategic goals. These are essentially quantifying the ability of the CBP to control and secure the Nation’s borders on the ground, in the air, or at sea, as well as the compliance with trade regulations at the border, the efficiency of cross-cutting tools and technologies enabling the collection and sharing of intelligence, and the quality of training of CBP personnel. Such measures of effectiveness for the first strategic goal include, but are not limited to, the number of land and coastal borders under control, the total number of border miles covered by permanent tactical infrastructure, the percentage of active ground personnel, of air support launches and marine support launches accomplished to secure the border, the percentage of strategically controlled high-risk border miles, of commercial containers and trucks inspected, and of apprehensions performed at the border. The measures of performance for the second strategic goal include, but are not limited to, the percentage of compliant passengers, and agricultural vehicles with law, regulations, and trade rules, the speed of clearance of passengers and cargo that do not require additional screening, the percentage of foreign suspicious or high-risk people and cargo examinations resolved, the total amount of shared information between the CBP and other concerned government agencies, as well as the number of patrol agents proficient in prevention, detection and investigation.

1.2.5. Threats to Homeland Security

The threats to homeland security are numerous and come in several forms [18]. They can be categorized as follows:

- **Biological**
    - Epidemic
    - Infestation
    - Public Health Emergency
    - Medical Emergency



- **Geological**
    - Earthquake
    - Landslide
    - Tsunami

- **Meteorological**
    - Cold Wave/Heat Wave
    - Drought
    - Flood
    - Storm Surges
    - Thunderstorm
    - Tropical Storm
    - Hurricane/Typhoon
    - Tornado
    - Snowstorm
    - Snow Avalanche
    - Freezing Rain
    - Water Main Break



- **Technological**
    - Industrial Accident (Hazardous Chemical Leak, Industrial Fire, Gas Explosion, Fuel Spill, Hazardous Material Spill)
    - Transport Accident (Train, Automobile, Aircraft)
    - Other Accident
    - Fire
    - Hazardous Chemicals Release


- **Man-induced**
    - Terrorism
    - Civil Unrest
    - Computer Hacking
    - Malicious Damage
    - Drug and Human smuggling
    - Contraband
    - Illegal Border Crossing

Therefore, the threats to homeland security can be classified into two main categories, namely natural and “man related.” In light of what has been presented so far, the present work focuses on “man related” threats. Among these, terrorism and illegal border crossing have been identified as the most rampant and, thus, deserve if not all, the major part of our most careful attention.

1.3. <u>Protection of Critical Assets as it is Organized Today</u>

In order to elaborate a strategy for surveillance, monitoring, and protection missions in the context of homeland security, it may be worthwhile to look more closely at how critical assets protection, including national borders control, is organized in the United States and in Europe, two of the major developed nations facing man-related threats to homeland security.

1.3.1. United-States vis-à-vis Europe

The September 11th, 2001 attacks on the World Trade Center and the Pentagon in the United States, along with subsequent terrorist attacks in European countries such as the United Kingdom and Spain, have prompted both sides of the Atlantic Ocean to reinvestigate their efforts to ensure homeland security and combat terrorism inside, at, and beyond their borders. However, European countries tend to approach homeland security, counterterrorism, and border protection differently from the United States.

1.3.1.1. <u>United States</u>

The American people have chosen to reorganize their domestic security and border protection institutions in order to include federal, state and local entities in the effort. This reorganization is characterized by the creation of a particular institution, the Department of Homeland Security (DHS), to provide the unifying core for the vast national network of organizations and institutions involved in the implementation of the national strategy for homeland security and counterterrorism. This strategy involves the cooperation and coordination of both public and private sectors playing a role in the protection of the homeland and the emergency planning of response to a threat, whether it be a natural disaster, a human error or a terrorist attack.

Indeed, federal, state and local officials have different roles in disaster response, homeland security and terrorism response situations. In natural disaster events, the federal government is responsible for early detection and forecasting activities. Federal agencies including FEMA, USACE and USDOT assist state and local governments in response and recovery operations. For homeland security and terrorist threats, the federal government is responsible for the detection and prevention of terrorist attacks, while state and local groups carry out preparedness and response activities. In order to be effective, disaster planning, response, recovery and mitigation activities must be fully integrated into "normal" planning and operational activities conducted in an inter-agency climate of cooperation and coordination. Disaster management represents a set of interdependent problems that require intensive communication and coordination among organizations and jurisdictions to reduce risk and losses [16]. For border protection, the U.S. Customs and Border Protection collaborates with federal, state, local, tribal, and international entities to enhance the security of both land and coastal borders while facilitating the legal travel and trade [17]. This layered approach involves several agencies, such as the Office of Border Patrol (OBP), the Office of Air and Marine (OAM), the Office of Technology, Innovation and Acquisition, and the Office of International Trade, that ensure that the U.S. borders are neither the first nor the last line of defense but one among many others.

1.3.1.1.1. Reference Documents

On July 16th, 2002, President Bush issued the "National Strategy for Homeland Security" [4], an overarching strategy for mobilizing and organizing the American Nation to secure the U.S. homeland from terrorist attacks. It communicates a comprehensive approach *"based on the principles of shared responsibility and partnership with*

Congress, state and local governments, the private sector, and the American people"—a truly national effort, not merely a federal one. The National Strategy for Homeland Security defines "homeland security" and identifies a strategic framework based on three national objectives.

In order of priority, these are:

- *Preventing* terrorist *attacks* within the United States
- *Reducing* America's *vulnerability* to terrorism
- *Minimizing* the *damage* and *recovering from attacks* that do occur

To attain these objectives, the "National Strategy for Homeland Security" [4] aligns homeland security efforts into six critical mission areas: intelligence and warning, **border and transportation security**, domestic counterterrorism, **protecting critical infrastructures and key assets**, **defending against catastrophic terrorism**, and emergency preparedness and response.

Then, the "National Strategy for the Physical Protection of Critical Infrastructures and Key Assets" [5], takes the next step to facilitate the strategic planning process for a core mission area identified in the "National Strategy for Homeland Security" – reducing the Nation's vulnerability by protecting our critical infrastructures and key assets from physical attack. It identifies a clear set of national goals and objectives, and outlines the guiding principles that underpin the efforts to secure the infrastructures and assets vital to the U.S. national security, governance, public health and safety, economy, and public confidence. It also provides a unifying organizational structure and identifies specific initiatives to drive the near-term national protection priorities and inform the resource allocation process. Most importantly, it provides a foundation for building and fostering the cooperative environment in which government, industry, and private citizens can carry out their respective protection responsibilities more effectively and efficiently.

Finally, the "National Border Patrol Strategy" [12] along with the "Secure Borders, Safe Travel, Legal Trade" [14] complement the "National Strategy for Homeland Security" by defining the critical role of the U.S. Customs and Border Protection in securing the Nation's borders – land, sea, and air – at major ports of entry and in between [15], while facilitating the movement of legal passengers and cargo in and out of the United States. The "National Border Patrol Strategy" identifies the major goals of the Border Patrol, lists the corresponding enablers, and summarizes the efforts required to take and maintain the operational control of the nation's borders, particularly that with Mexico and Canada. It builds upon many elements of previously enforced deterrence programs, such as Operations Gatekeeper and Hold the Line, but goes way beyond by incorporating six core elements: to deploy an appropriate combination of patrol agents, integrated detection and sensor technology, air and marine assets, and tactical infrastructure, to improve the rapid and efficient deployment of counteraction and interdiction units according to changes in smuggling routes and tactical intelligence, to develop operational control at and between interior check points, away from the physical border, to successfully deny illegal migration inside the country, to expand coordination and partnerships with other national or foreign law enforcement agencies to achieve better control of the borders, to improve the gathering, the sharing, and the exploitation of border intelligence and awareness, and to strengthen and centralize the structure of the command centers. In order to fulfill the priority mission of the Department of Homeland Security to prevent terrorists and terrorist weapons from entering the United States, the "National Border Patrol Strategy" focuses on the traditional Border Patrol's missions of detecting and apprehending illegal immigrants, drug and human smugglers, and other contraband dealers before they cross the borders. Finally, the "Secure Borders, Safe Travel, Legal Trade" completes "National Border Patrol Strategy" by addressing the issue of facilitating legal trade and travel while securing the nation's borders. It provides an overview of the main mission, the core values, and the vision of the CBP, as well as the challenges involved and the regional threats to the national security, the economy, and the public safety. It also gives a detailed description of CBP's main goals and objectives, and of the related cross-cutting enablers that are critical to CBP success in its double mission of expanding and maintaining operational control of the nation's borders while facilitating the flow of legal people and goods through the borders.

1.3.1.1.2. Organization

From an organizational point of view, in the United States, federal, state, and local entities have been combined into the Department of Homeland Security which mainly deals with border and transportation security, critical infrastructures and key assets protection, and defense against catastrophic terrorism, via its various affiliate offices and agencies. This is shown in Figure 1.

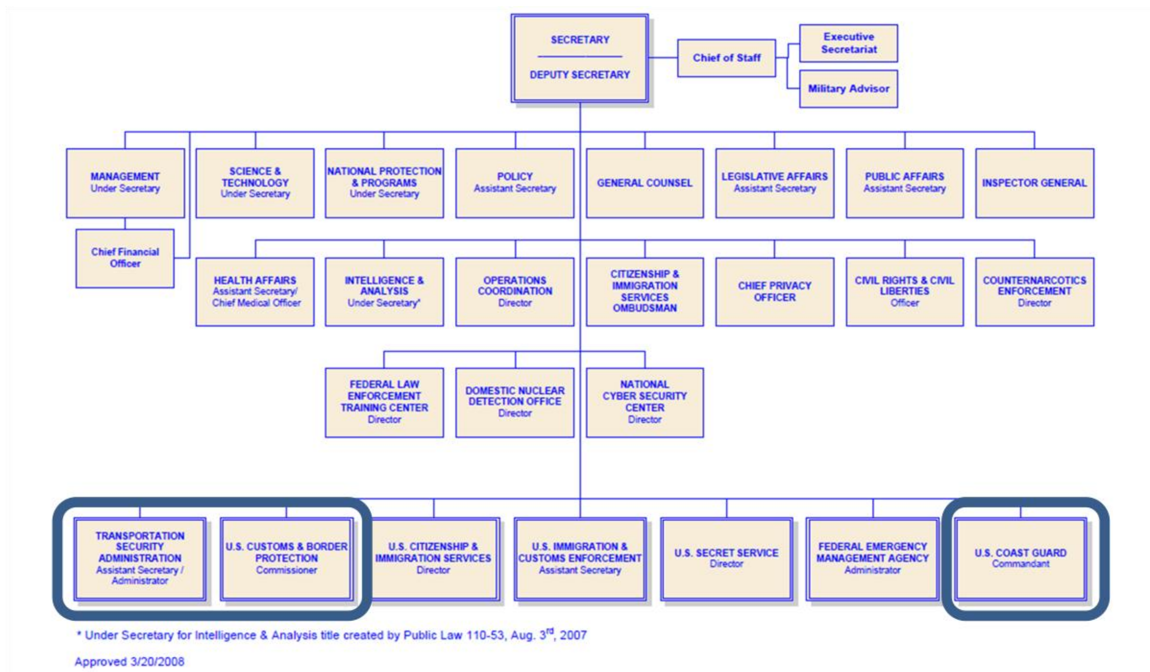


Figure 1: Organization of the Homeland Security in the United States [14]

In the above figure, the Transportation Security Administration (TSA), the U.S. Customs and Border Protection (USCBP), and the U.S. Coast Guard (USCG) collaborates to secure the nation's land, air, and sea transportation systems and borders while facilitating the flow of legal people and goods into and out of the United States.

In particular, the protection of critical infrastructures and key assets is organized around a variety of departments which have their own field of action corresponding to the sectors of critical infrastructures defined in the "National Strategy for Homeland Security" [4], such as agriculture, food, water, public health, emergency services, government, defense industrial base, information and telecommunications, energy, transportation, banking and finance, chemical industry and hazardous materials, postal and shipping. Figure 2 shows the federal government organization for the protection of critical infrastructures and key assets in the United States [5]. This is a centralized structure at the center of which is the Secretary of Homeland Security who coordinates the various departments, offices, and agencies responsible for securing the nation's assets.

President

Secretary of Homeland Security

Federal, state, local, and private sector coordination and integration

Comprehensive national infrastructure protection plan

Mapping threats to vulnerabilities and issuing warnings

Sector	Lead Agency
Agriculture	Department of Agriculture
Food:	
Meat and Poultry	Department of Agriculture
All other food products	Department of Health and Human Services
Water	Environmental Protection Agency
Public Health	Department of Health and Human Services
Emergency Services	Department of Homeland Security
Government:	
Continuity of Government	Department of Homeland Security
Continuity of Operations	All departments and agencies
Defense Industrial Base	Department of Defense
Information and Telecommunications	Department of Homeland Security
Energy	Department of Energy
Transportation	Department of Homeland Security
Banking and Finance	Department of the Treasury
Chemical Industry and Hazardous Materials	Environmental Protection Agency
Postal and Shipping	Department of Homeland Security
National Monuments and Icons	Department of the Interior

Figure 2: Federal Government Organization to protect U.S. Critical Infrastructures and Key Assets

In terms of border and transportation security, the U.S. Customs and Border Protection Agency is composed of several offices, each taking care of a particular aspect of border protection. For instance, the secure border initiative [13] and the secure freight initiative are meant to ensure the operational control of the border while allowing the legal movement of people and cargo through the national land and coastal borders. The above various offices are under the command of the chief of border patrol, the assistant commissioner of air and marine customs and border protection, and the assistant commissioners of intelligence and operations coordination, of field operations, and of international trade. This is depicted in Figure 3 which also highlights the importance of border protection through the secure border initiative and the chief of border patrol in apprehending illegal immigrants, drug and human smugglers, and other contraband- or terrorist-related individuals or groups.

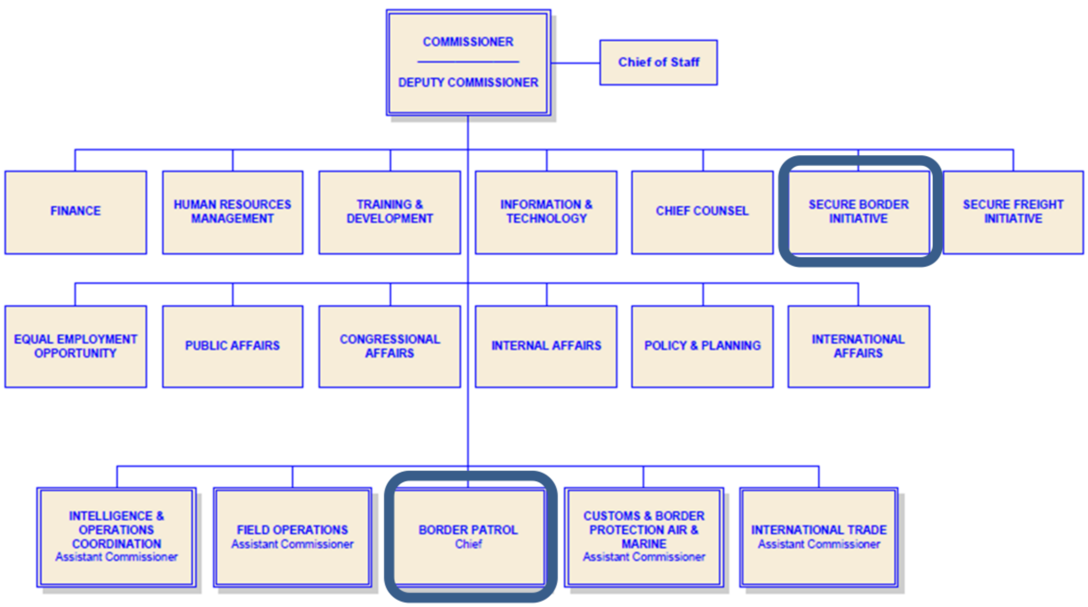


Figure 3: Organization of U.S. Customs and Border Protection Agency [13]

1.3.1.2. <u>Europe</u>

On the opposite side of the Atlantic, the European countries have decided to take advantage of their already existing institutional architectures to combat terrorism and respond to other security challenges and disasters, both natural and man-made. European countries have long been confronted to terrorist threats requiring military, law enforcement and intelligence responses. It is essential to recognize that the European experience with terrorism differs fundamentally from the threat of mass terrorism posed in recent years by terrorist organizations in the United States. In particular, European countries have been subjected to attacks by home-bred terror groups which are completely different in nature and in mentality from the contemporary transnational terrorist groups. These groups did not seek the destruction of a state or government, and were not really interested in causing major loss of life. Rather, they sought changes in policy or independence, and mass casualties would have compromised their cause.

However, it is legitimate to recognize that most European countries, with the exception of Spain and the United Kingdom (UK) which suffered from bombings in Madrid in 2004 and London in 2005, have not directly experienced the kind of catastrophic terrorism that has been directed against the United States. If they had, it is likely that they would now perceive the threat in a much different light. This may explain the differences in approaching the terrorist threat between the United States and Europe. For example, the Department of Homeland Security has no exact counterpart in European countries. As a matter of fact, the word "homeland" is relatively new in Europe, and the issue of homeland security has only been brought to attention in recent years. As such, most of the functions of the Department of Homeland Security are spread across a range of ministries and government agencies in Europe, such as the Ministries of Defense, Interior and Justice. Therefore, there is no entitled central entity that coordinates the efforts of homeland security in Europe, as is the case in the United States through the Department of Homeland Security.

1.3.1.2.1. Defending the Homeland

Homeland defense is part of the more general homeland security mission. As mentioned by John Clarke [19], "*The homeland defense mission, as set forth in doctrinal publications of the US armed forces, focuses on the role that military and other armed security forces play in ensuring the key elements of homeland security*."

Protection of populations and critical assets against potential harmful individuals, groups, or activities is one of the main issues in the homeland security rubric. This entails preventing people and cargo from illegally penetrating through the various land, air, and sea borders of the country via ports, airports, and permanent land ports of entry. In this area, European countries are almost unbeatable. As in the United States, there are critical infrastructures and key assets upon which European countries rely in order to function properly. These systems have become highly vulnerable over the past few years, even if they have gained in efficiency due to automation and electronic management. Nevertheless, they offer tempting targets for terrorists.

To face the terror movements of the 1970's and 1980's, European countries have devoted considerable resources to the protection of a broad range of potential targets that we now refer to as critical infrastructures. Therefore, European law enforcement and military forces have quite a developed experience in providing security for key governmental assets and individuals. In particular, the Ministry of Defense plays an essential role in the prevention of, and in the fight against, terrorism. For instance, in France, the armed forces, the national gendarmerie, the General Delegation for the Armament (DGA) (French Department of Defense), the Military Health Service (SSA) and the intelligence agencies under the Ministry of Defense are permanently in alert to secure the French population and the critical assets they depend on. "*Defense has the mission to ensure the security and the integrity of the territory as well as the population safety, any time, under any circumstances and against any attack*" [20].

European Ministries of Defense and associated agencies are equipped with specific capabilities to ensure the security of the population in every situation, along with the well-being of the society. For example, in France, the protection of the population against chemical, biological, radiological, nuclear and bacteriological threats is made possible through the expertise of hundreds of scientists working in the laboratories of the General Delegation for the Armament (DGA) or in the Military Health Service (SSA). Other missions incumbent on the Defense sector include securing maritime and aerial routes and borders by employing state-of-the-art technology, gathering intelligence, supporting the population in the organization of special events, providing emergency medical assistance, controlling supply flows, protecting critical installations, and planning and managing complex crisis situations.

Actually, as shall be detailed in subsequent sections, the Defense sector in Europe is the only public institution capable of deploying a rapid and strong response to face a particular situation, not only in terms of strength, but also in terms of volume and power of the technologies used in the field, and in terms of logistics and organization of the activities. The protection of the national territory, of the population and of the interests of the country is at the heart of the mission of the Ministry of Defense and related agencies. The emergence of a mass terrorism has made homeland security a priority. It is now all about protecting the nation's borders, critical assets and population against aerial and maritime threats, securing sites, protecting freedoms and continuity of government action by ensuring the delivery of essential services for the safeguard of the population and the well-being of the society [10].

Besides, Europe has considerable experience in intelligence sharing as part of a counter-terrorism effort: in France, the "Secrétariat Général de la Défense Nationale" (SGDN) coordinates the efforts of the judicial and executive branches. European countries are facing the same concern as the United States as regards the implementation of a strategy improving security, but still compliant with civil liberties. While there has been vigorous debate on such issues in the United Kingdom, civil liberties are fervently and traditionally protected in Germany, Spain, and Italy in light of their histories. Nevertheless, due to recent terrorist activities, such as the terrorist attacks on commuter trains in Madrid in 2004 or the metro bombings in London in 2005, these countries are currently discussing the strengthening of measures destined to ensure domestic security. As to France, even if such values as free speech and freedom of religion are still vigorously protected, the need for public order that emerged after the September 11th, 2001 attacks in the United States has progressively lead the French society to take steps to enhance security [21].

When it comes to the budget allocated to intelligence and law enforcement efforts against terrorism, it varies widely among European countries, although most of them have devoted increased funds to such efforts over the last years. For instance, the European commission allocated 15 Million Euros of funding to 15 projects in the field of security between 2004 and 2006. The projects were funded under the Preparatory Action for Security Research (PASR), which focuses on bridging the gap between civil research and national security research initiatives. The security issues addressed by the funded projects go from the detection of explosives and biological agents to the development of tracking and surveillance systems [22],[23]. Between 2004 and 2007, the PASR has funded a total of 39 security research projects to the tune of 45 Million Euros. Indeed, security research is a key part of the Seventh Framework Program or 7FP with an average annual budget of 200 Million Euros. The European commission has also prepared a program for the "prevention, preparedness and consequence management of terrorism." The funding is supposed to be approximately 1,400 million Euros "sufficient for soft issues." However, the commission considered making available an annual amount of 250 million Euros between 2007 and 2013 for research into practical strategies for risk mitigation [24] under the Security Work Program. The later program is strictly focused on the implementation of policies to support civil applications of security in the domains of transport, health, energy, environment, and population protection. The primary mission areas funded under the FP7 Security program encompass the protection of citizens against terrorist activities and organized crime, the protection of infrastructures, potential target sites of political or symbolic value, and utilities such as energy, transport, communication, and finance, the surveillance of the borders against illegal entries of people and goods, the enhancement of emergency management, recovery, and rescue tasks in case of a security crisis, the improvement of the integration, intercommunication, and interoperability of security systems, equipments, services, and processes, as well as of the standardization and sharing of intelligence information and processing, the study of the socioeconomic aspects of security measures such as the attitude of citizens, the behavior of crowds, the communication issues with the public, the public acceptance and understanding of security and control measures, and the analysis and building of scenarios to coordinate and structure security research [25]. Nonetheless, since responsibilities for homeland security issues are often spread among different government entities, it is difficult to evaluate and compare funding allocated to measures for strengthening transport security, improving emergency preparedness and planning, countering chemical-biological incidents and protecting populations and critical assets among European countries.

This is especially true in the domain of border protection, where two separate entities, the Customs Administration and the Border Guards Administration, share responsibilities in managing the borders of the European countries [26]. Indeed, depending on the country, the form and extent of cooperation between the two agencies vary significantly, from being completely independent and lacking communication, through having a substantially developed cooperation, to being merged into a single entity. This diversity in handling border protection from one European country to the next essentially comes from the differences in institutional set-up, legislation, and historical powers and competencies of both Customs and Border Guards Administrations. More specifically, the main challenges to cooperation between the aforementioned administrations can be divided in three classes. The first class concerns operational challenges such as legal obstacles related to privacy, sharing of financial planning, equipment, infrastructure and costs, trade and intelligence data protection and sharing, as well as technical incompatibility or mistrust between the two entities. The second class consists of challenges related to differences in institutional cultures, and to functional and organizational inequalities which produce tension between the two agencies and prevent them from cooperating and communicating efficiently. For instance, on the one hand, Customs agencies in the European Member States such as the Civil Guard Corps in Spain, the Guardia di Finanza in Italy, the Gendarmerie Maritime in France, and the Coast Guard in Greece are part of the Finance Ministry which has the institutional culture of a civil administration. On the other hand, the Border Guards and other Law Enforcement Agency Institutions performing similar duties often belong to the Ministry of the Interior or to the Ministry of Defense whose organizational structures has retained some form of military culture. This generates significant differences in the models of recruitment, education, training, and carrier paths of both Customs and Border Guards officers, so that their competencies used to exclude any overlap of tasks and functions and cooperation is mostly based on informal contacts. The last class of challenges concerns political considerations. Political influence on border protection is felt differently depending on the European Member State considered: it can be inexistent at all levels of command, it can be present at the highest level of command only, or it can act directly via the election of the Heads of Customs and of Border Police. This political influence presents two main obstacles to the cooperation between the Customs and the Border Guards Administrations. First, the loyalty of Customs to the Ministry of Finance, and that of the Border Guards to the Ministry of Defense generates suspicion and avoidance of cooperation with the competing ministry. Second, changes in the organizational structure of the agencies tend to make it difficult to maintain informal contacts and trust developed between their managements in the long run [26].

1.3.1.2.2. Military Civil Support and Forces for Homeland Security

Europe has a broader experience concerning missions involving military support to civil authorities: the restrictions are less definitive and the range of operations is larger than in the U.S. Many countries, such as France and Germany, do assign military forces specific authority to support civil authority. On the contrary, in the United States, the military forces are always in a supporting role, generally restricted to executing missions of military assistance to civil authorities.

In many European countries, the use of military forces is not considered a last resort. Rather, their employment is often considered a matter of course. As such, military forces have broader responsibilities than in the United States: they directly support civilian authorities in times of disaster, whether natural or man-caused. In the United States, presidential authority is required for the employment of active-duty forces, whereas in Europe generally much simpler requirements apply. Many European armies can be directed to engage in support tasks by the defense minister, rather than the president or the prime minister. Hence, it is not uncommon to find military units supporting international events, such as bicycle and ski races, as well as large cultural events such as major exhibitions and fairs. In this European context, military forces have been employed so often that their support to civil authorities during major disturbances is relatively uncontested and even taken for granted [19].

Contrary to the United States, European countries do not have a federal structure and decentralized law enforcement. Instead, they have considerably more options at their disposal for carrying out key homeland security tasks. Indeed, in addition to military forces, there are many different kinds of police forces playing a significant role in homeland security that do not have any equivalent in the United States. Unlike the United States, European countries have long employed active-duty forces and reserve forces in executing key homeland security missions, and still maintain a large number of military formations on their home territory. Consequently, they have a high density of armed forces relative to population, and these are available for homeland security missions.

In addition, many European countries have specialized police forces at their disposition, capable of a broad range of homeland security functions. In particular, the paramilitary forces have long experience in carrying out tasks associated with combating terrorism such as infrastructure protection and special event security. Forces like France's Gendarmerie, Italy's Carabinieri and Spain's Guardia Civil represent force models nearly ideal for homeland security missions due to their high mobility, equipment and experience in law enforcement activities.

Finally, to complement these paramilitary forces, several kinds of police forces also carry out homeland security tasks. For example, in addition to municipal and local police forces, many European countries have national police forces, such as Austria's Gendarmerie, composed of small, highly specialized units responsible for domestic counterterrorism operations. In France, the "Compagnies Républicaines de Sécurité" function as the principal reserve of the national police force and are frequently employed in special event security tasks and critical infrastructure protection.

1.3.1.2.3. Reference Documents

As part of the critical infrastructures and key assets protection, the "Critical Infrastructures Protection as part of the fight against Terrorism" [18], defines the major threats faced by the European critical assets and lays out clear suggestions about how to improve European prevention, preparedness, and response to terrorism involving critical infrastructures. In the "Prevention, Preparedness and Response to Terrorist Attacks" [27] and the "EU Solidarity Programme on the Consequences of Terrorist Threats and Attacks" [28], the European Commission proposed a European Programme for Critical Infrastructure Protection (EPCIP) [29] and a Critical Infrastructure Warning Information Network (CIWIN) [30] in order to homogenize the procedures and rules for the protection of critical assets in all Member States of the European Union.

As part of border protection, the Laeken European Council of 2001 adopted the concept of "integrated management system for external borders" which states that the issues of terrorism, illegal immigration, and drug and human trafficking can be addressed more efficiently via better coordination and management of all the activities performed by the public authorities of the Member States to efficiently control Europe's external borders [31]. In this context, the Seville European Council of 2002 approved an "Action Plan on the Management of External Borders of the European Union" which defines legislative and operational measures for the collaboration of agencies on the national level. These measures are mainly based on a Commission Communication of May 2002 on integrated border management [32] and on a feasibility study of May 2002 on a European Border Police. In 2005, the European Council adopted the Hague Programme [33] to structure the objectives of action for the following five years and to define the development of a second generation set of measures to strengthen the control of the Union's external borders. The Hague Program promotes coordination both at the European level and at the national level between law enforcement agencies such as the Police, the Customs, and the Border Guards, so as to reach optimal levels of protection. In actuality, cooperation of customs agencies at the European level has developed independently since the foundation of the customs union in 1968 and of the Community Customs Code (CCC) in 1992 which defined a uniform scope, as well as rules and procedures for border control at the European Union level [26]. In particular, the 2005 amendments of the CCC introduced a layered approach to the movements of goods across international borders through the World Customs Organisation Framework of Standards to Secure and Facilitate Global Trade (SAFE Framework) at the EU level which has to be equally and fully applied in all the Member States [34]. Other provisions on cooperation in customs criminal enforcement ("third pillar") have been added to those laid out by the Customs Union after the Amsterdam Treaty of 1997 [35]. Then, the Vienna Action Plan of 1998 [36] ratified the Convention on mutual assistance and cooperation between Member States (Naples II) [37] and the Convention on the use of technology for Customs purposes (CIS Convention) [38] in order to specify the goals of the "third pillar" customs cooperation. In 2003, the Commission focused on the cooperation between Customs and other border agencies to integrate their border management practices, to develop common risk analysis schemes, and to exchange intelligence information and apprehension data. These ideas are supported in the "Strategy for the Evolution of the Customs" [39] of 2008 where coordination and cooperation of the Customs with other law enforcement agencies with similar duties are highlighted as major components of the evolution of the customs administrations at the European Union level. Since 1999, when the Amsterdam Treaty went into application, the European Council on Justice and Home Affairs had been trying to strengthen the cooperation in the area of migration and security of people and goods across Europe's external borders. In 2002, this led to the creation of six centers of command to form the External Border Practitioners Common Unit tasked with overseeing European-wide projects and operations related to border management. Two years later, in 2004, the six ad-hoc centers – the Risk Analysis Centre in Helsinki (Finland), the Centre for Land Borders in Berlin (Germany), the Air Borders Centre in Rome (Italy), the Western Sea Borders Centre in Madrid (Spain), the Ad-Hoc Training Centre in Traiskirchen (Austria), the Centre of Excellence in Dover (united Kingdom), and the Eastern Sea Borders Centre in Piraeus (Greece) – were complemented by the establishment of agency for the Management of Operational Cooperation at the External Borders of the Member States of the European Union (Frontex) [40]. Although the European Union has defined measures for cross-border and external-borders cooperation between Member States and between Member States and the Commission, it has still not regulated the inter-agency collaboration between Customs and Border Guards at national levels [26]. Some preliminary regulations or recommendations have been adopted but a number of factors are adversely affecting police and customs cooperation in many Member States. These factors have already been explored in the previous sections and mainly concern the different nature of the procedures, of the institutional cultures, and of the backgrounds, the competition between the agencies, and the detrimental absence of a strategic approach to the issues of border management.

1.4. <u>Focused Problem</u>

This work is performed as part of a more general question about the methods and tools useful in organizing the protection of borders, critical infrastructures, key assets and populations against illegal activities and terrorism. The surveillance of geographic borders and critical infrastructures using limited sensor capability has always been a challenging task in many homeland security applications. Although threats to homeland security can be conceived to take place in many forms, those regarding illegal penetration of the air, land, and maritime domains under the cover of day-to-day activities have been identified to be of particular interest by some U.S. and European governments. For instance, the proliferation of low altitude aerial systems, combined with regular air traffic growth, poses a unique challenge for the surveillance of homeland airspace and in particular for identifying potentially hostile vehicles interoperating with friendly aircraft. Similarly, the proliferation of drug smuggling, illegal immigration, international organized crime, resource exploitation, infectious diseases, environmental degradation, and more recently, modern piracy, require the strengthening of land border and maritime awareness. Hence, land border and maritime intelligence assessments point to increasingly complex and challenging national and coastal security environments. In this context, the ability to monitor, collect information (i.e. detect, identify) and eventually intercept suspicious entities or systems well before they reach the border or strategic land and coastal sites is of critical importance to prevent dangerous activities from jeopardizing populations and governments.

1.4.1. The Wake up Call

The terrorist attacks of September 11th 2001 constitute the most striking example of the proliferation of a kind of terrorism that uses systems developed by humans to make their life more pleasant and enjoyable, to their own detriment. On that tragic day of September 11th 2001, leaders of governments from all around the world as well as millions of private citizens were awoken from their slumber of national security and safety. Since that catastrophic and unforgettable event, people living in societies qualified as “developed” have realized how their life could turn out to be, should a major attack on critical assets happen. Reliance on many basic necessities such as electricity, water or food, is so high that a large disruption in those services could turn day to day life into a nightmare. Can you really imagine surviving without water, electricity, home heating oil, natural gas, automobile gasoline, telephones, Internet access, emergency services, etc? [41].

The attacks proliferated in 2001 in the United States raise the issue of efficiently detecting and identifying potential malevolent systems, entities or activities among widely accepted systems, behaviors or procedures, before they cross the land, air, and sea borders and take place in the country, in order to protect human life and property. Indeed, it cannot be assumed that attacks on the scale of the September 11th tragedies will not be repeated one day, or that the attacks will be similar in terms of instruments and targets. Clearly, the anthrax attacks that followed demonstrated that the threat goes beyond specific terrorist networks and the use of hijacked aircraft as weapons.

Finally, one can notice that, beyond the political objectives that may motivate the use of terrorism, the principal motivation is the creation of fear. The psychological impact is one of the main reasons for terrorist-like violence, such that over-responding to an attack is precisely what terrorists are looking for: over-reacting to an attack would give it full credit by exaggerating the threat and its consequences.

1.4.2. A Widespread Disease

As mentioned earlier, illegal movement of people and goods through national air, land and marine borders has now become a widespread issue as part of the general fight against terrorist activities. In particular, aerial systems flying at low altitudes, drug and human traffickers, and illegal contraband smugglers have become rampant items of interest in recent years. This may be explained by the quasi-inability of detection and warning systems able to properly detect, identify, and classify the above items of interest in a timely fashion. With time, the later have demonstrated the ability to exploit vulnerabilities and know how to adapt to changes in security measures applied to both Air and Border Protection. For instance, the exploitation of the Air Domain by terrorists and hostile nations using unconventional and sophisticated attack methods is not a recent phenomenon.

In the years following the September 11th attacks, security in the aviation sector has been significantly strengthened in many countries over the world, especially in the United States. For example, measures taken in the United States include: *"a federalized Transportation Security Officer workforce that screens passengers and baggage traveling on passenger aircraft; hardened cockpit doors to prevent unauthorized access to flight deck; Federal Air Marshalls who fly anonymously on commercial passenger aircraft to provide a law enforcement presence; enhanced explosives and threat detection technology deployed in hundreds of airports; airspace and air traffic management security measures; and a cadre of canine explosives detection teams screening baggage, cargo, and increasingly, carry-on items"* [43]. In addition to these measures, expanding air, land, and marine surveillance and interdiction efforts have been put into place to counter terrorist threats, and traditional defense activities against threats from hostile nation-states continue to be performed.

Threats to the homeland security are numerous, complex and adaptive, essentially because of globalization, technological advances, proliferation of weapons of mass destruction and emergence of terrorism as a global phenomenon. They can be analyzed in two ways depending on the originator and on the targets and tactics:

- Terrorists groups are a kind of originators. They are politically, as well as in some cases religiously, motivated. They mostly use premeditated violence to affect a particular audience and give a greater impact to their. For instance, their ultimate goal in the Air Domain is to conduct multiple, simultaneous, catastrophic attacks exploiting the Aviation Transportation System because it is being interpreted as a symbol of the global presence and economic influence of modern developed nations.
- However, the terrorist threat is changing in form and intensity as terrorists' intentions and capabilities change and countermeasures are instituted. They are adapting their tactics and techniques on multiple fronts (e.g. planning, complexity of attack and style of execution) to exploit vulnerabilities in the system. They also choose the type,

location and frequency of attacks depending on their perception of the level of security of a target at that period in time, in the current context. That is why such parameters as type, location and frequency of attacks cannot be extrapolated from historical patterns and therefore, current threats must be regularly reassessed.

1.4.3. The step Towards a Cure

To counter all these threats to the protection of the homeland and especially the ones willing to illegally penetrate the nation's border to harm critical assets or populations, advanced warning is incontrovertible since it grants time and distance to counter adversaries, whether they are planning an operation or are en route to commit an attack or any other unlawful act.

Needless to say malevolent individuals or entities will continue to exploit the global air traffic growth and the globalization of trade to threaten the Nation's critical infrastructures, key assets and population. Nations must therefore continuously monitor and exert unambiguous control over the access to its airspace, and its land and coastal borders. Enhanced surveillance coverage coupled with security measures, collection, sharing and efficient dissemination of intelligence information, as well as response capability, will allow the nation to seize initiatives and influence events before adversaries can cause harm to people and property.

Furthermore, to achieve all of the strategic objectives including deterring and preventing terrorism, illegal immigration, smuggling and contraband, as well as protecting the Nation and its interest, or mitigating the effects of any unlawful act, it is indispensable to maximize global air, land, and sea awareness. Nevertheless, this heavily dependents upon advanced information collection and unprecedented cooperation and action among various elements of the public and private sectors, both nationally and internationally. At the same time, all such procedures and actions still have to be compliant with laws protecting civil liberties and free flow of people and goods, which complicates the matter further.

Additionally, air traffic growth, technological advancements in aircraft design, and development of air tourism cause the emergence of new challenges such as the detection of stealth aircraft, or aircraft flying at low altitudes or following terrain features very closely to mask their presence. It is then crucial to maximize the Nation's capability to detect and monitor aircraft within its airspace and its contiguous areas, from large commercial aircraft to low-altitude, low-observable manned or unmanned aircraft. Similarly, with the development of global commerce and tourism, border protection becomes cumbersome. It is now all about being able to detect and identify people or products that could harm the Nation as they hide among the common flow of people and goods at the country's ports of entry.

Another important issue is the monitoring of those aircraft, cargo and persons of interest from the point of origin, throughout their aerial, ground, or maritime route, to the point of entry, so as to ensure the integrity of the transit, manage aviation or maritime traffic routing, and, if necessary, interdict and/or divert aircraft, ground vehicles, or marine vessels for law enforcement or defensive action. The aforementioned missions can be leveraged by the development and, where appropriate, the implementation of new and emerging technologies including both airborne and ground-based systems for detecting potential items of interest, as well as for reducing susceptibility and vulnerability to illegal immigration, smuggling, and contraband, and consequently to terrorism. In this effort, it seems worthwhile to conduct comprehensive assessments of threat, likelihood, vulnerability, and criticality to identify security measures that require improvement, as well as to develop a consistent risk management approach.

The objective of homeland security is to protect the people, physical entities and cyber systems that are indispensably necessary for survivability, continuity of operations and mission success, as well as to deter or mitigate attacks by people (terrorists, hackers,…), by nature (hurricanes, tornadoes,…), and by HazMat accidents. Nonetheless, it is impossible to prevent all attacks against the whole nation. Hence, applying a methodology to protect people and infrastructures can reduce the chances of future attacks, make it more difficult for terrorists to succeed in their objectives of degrading infrastructures or causing mass casualties, and mitigate the outcomes when they do occur.

Protection of the air, land, and marine borders as part of the protection of populations, and critical infrastructures and key assets is meant to be proactive, preemptive and deterrent in nature, having in mind the will to change the behavior of those willing to do evil: the proper protection might have the potential to develop a new “mindset” among the later that their action will be futile and not yield the results they seek [44].

The ultimate objective of homeland security is thus to deter future attacks on our homeland by convincing terrorists and other perpetrators of illegal acts that their action will not succeed or that our response will cripple their cause. As such, the strategy mantra for the future should be the power of balance rather than the balance of power. This means that deterrence in the 21st century will require an evolving suite of operational capabilities that hedge our bets against thinking adversaries equipped with an infinite array of asymmetric weaponry. The approach suggested to tackle 21st century terrorism involves deterrence, prevention, preemption, crisis management, consequence management, attribution, and response. Deterrence is twofold: denial and punishment. Prevention is most effectively accomplished through layers of defense to deny adversaries from acquiring materials, equipment, intelligence or knowledge that would enable them to create or deliver items that could harm the nation’s populations and critical assets. The present thesis work is intended to address the aforementioned last point of the preventive action. Namely, the goal is to define detection architectures of both fixed and mobile sensors able to protect specific critical assets in various operational environments through early and accurate detection of potential threats.

Nevertheless, the dilemma, in such cases, is always to decide how much protection is enough, that is, to effectively exploit the power of balance. An optimal system will integrate a nationwide array of preventive actions, whose sensitivities are adjusted in near real time according to the level of threat, and the synthesized sensory data. This enhances tactical situational awareness, essential to denying the adversaries achievement of the desired impact, while informing an evolving assessment of asymmetric threats to the homeland [45].

To summarize, in order to preserve life and to minimize the risk to public safety, it is required to prevent terrorist attacks, criminal activities, hostile acts or unlawful exploitation of legal trade and travel. Indeed, as strongly highlighted, aerial, ground, and maritime border crossings have become highly suitable for the proliferation of means to approach not always enough, but supposed protected, yet vulnerable, assets. Then, sound and timely decisions about, and response to, the full range of aerial, land, and maritime threats need to be supported, to enable shared situational awareness of the homeland. This integrates intelligence, surveillance – including sensor inputs, reconnaissance – and other useful information, including information on other critical infrastructure elements such as potential ground targets [9].

1.5. <u>The Missing Piece</u>

1.5.1. Strategic Challenges

As mentioned earlier, our populations and way of life are the source of great strength, but also a source of inherent vulnerability. Our populations are large, diverse and highly mobile, allowing people willing to perform terrorism or any other illegal activities to hide within our midst. The organization of our societies makes people congregate at schools, sporting arenas, office buildings, concert halls, high-rise residences, and commercial malls, presenting targets with the potential for mass casualties and facilitating illegal transport of drugs, humans, and other contraband [4].

Thus, a key challenge faced by the defense and protection community is to be able to properly detect and identify potential harmful or illegal people and activities, as they blend into the daily routine of legal trade and travel. A French government agency summarizes the focus of the defense community in Europe as follows: "*During different types of conflicts [and even during peace] it is not only relevant to perform the detection, but also to evaluate the air situation to distinguish between friend and foe and to determine a ranking in severity of threat*" [46]. For instance, aerial systems flying at low altitudes could turn out to be potential threats to critical assets due to their rampant proliferation and the quasi-absence of systems able to detect and identify them properly in a timely fashion. "*The Air Domain serves as the medium for a variety of threats that* *honor no national frontier and that seek to imperil the peace and prosperity of the world*" [43]. The most unsettling observation is that many of these threats hide among legitimate systems. They take advantage of regular air traffic growth, general aviation, and leisure travel, either to carry out hostile acts, or to make available illegal merchandises, or even weapons of mass destruction, related materials or their delivery systems to hostile individuals or groups willing to disrupt the national security, the economy, the public safety, and the critical infrastructures and key assets.

Thus, the European Defense Community is convinced that there is an undeniable need for enhancing the capabilities of detection and intervention against those that could be a threat to the Nation's homeland. In such a context, anticipation and flexibility are the main characteristics. Anticipation is made possible by the collection of intelligence information beforehand and is based on the cooperation of several entities at the local and international levels. Flexibility is ensured by the mobilization and coordination of a number of means of intervention scattered across the territory, or by special units of operation. Moreover, it is necessary to enhance the protection of borders, critical assets and populations by deploying efficient surveillance and response equipments able to prevent and mitigate illegal or harmful actions before they happen or as they are unfolding. Also needed is a broadening of the database on means and capabilities made available to the military forces to protect populations against illegal or terrorist acts [10].

In other words, in the particular context of homeland and airspace protection, the issue is to determine the adequate types and positioning of ground-based detection systems that provides adequate surveillance of all borders, critical assets and populations. The inherent challenge in designing and optimizing a robust detection and protection architecture of sensors is further compounded by the complex nature of the systems-of-systems required to provide efficient and effective coverage of the whole aerial, land and maritime area surrounding the border or the critical asset of interest, as well as nearby populations.

Furthermore, the wish to protect all geographic borders and critical assets using limited sensor capability is usually hindered by the length of the border or the nature of the area it runs through, as well as by the size and number of the critical assets and the populations living around. As a result, it is virtually impossible to secure both each and every mile of border around the country, and each and every critical infrastructure inside the country. Most often, a compromise must be made between the percentage of border or critical asset covered by surveillance systems and the induced cost. This is also valid when trying to satisfactorily and economically protect a densely populated area where many lives are at stake and where detection of suspicious and potentially harmful activities is highly compromised due to the nature of the surrounding environment.

Another challenge faced by the defense and protection organizations is the evolutionary nature of the notion of criticality in the determination of the assets to protect and in the way to defend the borders. Criticality varies as a function of time, risk and market changes. Acting to secure the highest priority facilities, systems and functions, one should remain aware that adversaries can shift their interest to less protected targets that are more likely to yield desired shock effects, or can adapt to the current state of protection and invent new ways to counter it.

Finally, several forms of data are crucial for the decision makers to make an efficient and effective choice as regards the protection architecture. Data must be available as soon as possible. Although they can be provided by several types of sensors deployed on-site (such as radars, optronic systems, humans, etc), investigating, verifying and using the data in real-time is a nontrivial task. Plus, the data required for early detection and warning in real-time are perishable and must be updated and verified regularly.

Another issue stems from the fact that the state of technology of critical assets, which particularly impacts their vulnerability, might be rapidly outdated due to the ever increasing modernization of security requirements for such structures. Hence, the level of protection required for a critical asset today might be different from what it will need to be in a few years. Similarly, the technology used to protect borders from illegal entries of people and goods today might become obsolete in a few years as people respond to the current state-of-the-art by developing new means to perform their illegal or harmful actions. As such, the defense and protection community needs to find a way to forecast the future state of the world, as well as the corresponding level of protection required, depending on the current vulnerability to specific threats [47]. The prospective plan has been established to forecast the potential future states of the world and to design adequate systems to tackle emerging challenges. Now, there is a need for a methodology enabling the design and optimization of a protection architecture, able to merge newly developed systems among themselves, or with existing systems, in an effective manner, in order to protect borders and critical assets in an environment of persistent and evolving threats. In a systems-architecture study, many interactions between the components of a system-of-systems are highly interdependent and interrelated. Besides, the system-of-systems nature of the problem involves handling a large number of variables which can go from the system level to the environmental and human factors, and thus be highly heterogeneous. Another problem incumbent to system-of-systems studies is the integration and the reconfiguration of system interactions with the emerging of new or existing systems with time.

However, to date, a structured methodology for the design of network-enabled systems does not exist: communications pathways and sensor attributes are often held constant in systems studies, when in fact they may be the dominant design drivers for future systems-of-systems. Hence, a methodology that addresses network connectivity and the selection of systems' attributes must be infused with existing techniques for platform sizing and synthesis.

Therefore, the envisioned goal for the proposed research is the development of a high-level, robust and adaptable methodology, *"a set or system of methods, principles, and rules for regulating a given discipline"* [48]. This methodology will be developed with the problem of border and critical assets protection in mind, but is intended to be sufficiently general to be applied to a range of problems in the fields of systems-of-systems engineering and distributed sensor networks.

Additionally, the application of this process to the protection of borders and critical infrastructures and key assets is intended to result in the creation of a parametric tradeoff environment which would synthesize the data and show relevant information to decision makers. This preliminary framework is meant to demonstrate the feasibility of a more complex environment capable of grasping the whole extent of the problem of distributed sensor networks.

1.5.2. Research Objectives

The primary objective for this work is the development of a structured methodology that supports the generation of a prioritized portfolio of ground-based detection architectures, for the protection of national air, land, and sea borders, as well as of critical infrastructures, key assets and populations. For each operational scenario considered, the idea is to determine the optimum combination of type, number, position, and design characteristics of each constituting detection system in the architecture. A key challenge in developing such a methodology lies in the integration of multiple heterogeneous elements that comprise a "system-of-systems" and that must work together or interact to provide a desired capability.

With this in mind, the overall research objective can be summarized as follows:

The focus of the proposed study is the development of a structured, traceable, reproducible and practical methodology, addressing the multi-criteria design, Modeling, Simulation, and Optimization of Distributed Detection system Architectures (M-SODDA) in the context of homeland security applications. On the one hand, the M-SODDA methodology is intended to facilitate a quantitative assessment of the operational and technological potential of optimized protection architectures, with respect to capability-level measures of effectiveness. On the other hand, the M-SODDA methodology is meant to allow a rigorous analysis of the design and optimization factors enabling the decision maker to explore the design space and assess the relative sensitivity of tradeoffs at the system and subsystem levels.

Due to shortcomings in current methods for resource allocation, the proposed methodology needs to be "structured, valid, defensible and practical" and thus implies a "quantitative assessment." A structured methodology that relies on the physics of the problem is a way to quantify the benefit of technologies and to enable more informed decisions. The term "technological potential" refers to the ability of a technology to meet precise high level requirements through the possibility for further development or, more precisely, the possibility for further improvement in the technology attributes relative to inherent limits. The term “capability” refers to “*the ability to undergo or be affected by a given treatment or action*” [48] or to the ability to achieve an effect.

Finally, another objective of this research is to cut through the complexity of systems-of-systems, namely to integrate large and disparate volumes of data, and to filter and display the resulting relevant information, in such a way to aid the decision maker to make informed decisions earlier in the design process [49]. To formulate a successful approach, however, it is first necessary to review existing methods for designing, modeling, simulating, and optimizing detection systems-of-systems or other distributed sensor networks, in order to identify whether current techniques address the needs for resource allocation and network connectivity inherent to systems-of-systems studies.

The focused problem is then the multi-criteria modeling, simulation and optimization of complex, multi-disciplinary detection architectures for the protection of air, land, and sea borders, and of critical infrastructures and key asset in the contexts of homeland security and public safety. In summary, what is of interest for this research is the detection of illegal or harmful individuals, groups of individuals or activities.

CHAPTER II

BACKGROUND RESEARCH

A review of theory and current literature is given before describing the various steps of the M-SODDA methodology and going into more details about its implementation. The discussion is qualitative for now. The following theoretical concepts are reviewed:

1. Systems-of-systems
2. Systems engineering and systems-of-systems engineering
3. Parametric analysis
4. Modeling and simulation
5. Optimization

This chapter serves as the basis for the construction of the research questions and hypotheses discussed in the next chapter.

2.1. Introduction to Systems-of-Systems

"Everything should be made as simple as possible, but not simpler."

- Albert Einstein

While systems have been widely studied over the last several years, there still exists a lack of proper methods to deal with systems-of-systems. The subsequent sections introduce a terminology for systems and systems-of-systems, and delineate the frontier between a system-of-systems and a system architecture for the purpose of defining baseline definitions in the context of the present research.

2.1.1. What is a System?

The word system, which comes from the Greek “sunistanai” meaning “to combine,” is defined in the Department of Defense (DoD) dictionary [50] as “*A functionally, physically, and/or behaviorally related group of regularly interacting or independent elements; that group of elements forming a unified whole.*” The DoD further describes a system as “*A collection of components organized to accomplish a specific function or set of functions.*” In the same vein, the International Council on Systems Engineering (INCOSE) [51] considers a system as “*A combination of interacting elements organized to achieve one or more stated purposes.*”

The Institute of Electrical and Electronics Engineers (IEEE) [52] more broadly specifies a system as an “*Interdependent group of people, objects, and procedures constituted to achieve defined objectives or some operational role by performing specified functions. A complete system includes all of the associated equipment, facilities, material, computer programs, firmware, technical documentation, services, and personnel required for operations and support to the degree necessary for self-sufficient use in its intended environment.*”

Finally, the SMC Systems Engineering Primer & Handbook [53] summarizes the previous definitions in one sentence, namely “*A system can be thought of as a set of elements which interact with one another in an organized or interrelated fashion toward a common purpose which cannot be achieved by any of the elements alone or by all of the elements without the underlying organization.*”

2.1.2. Categories of Systems

In the light of the aforementioned definitions, the system’s structure can be divided into three different groups: simple, complicated and complex, each corresponding to specific properties.

Simple systems can be defined as isolated systems in their dynamics and their observation: they can be reduced to a set of weakly coupled one-component sub-systems, each of them exhibiting dynamics that are mainly ones of isolated elements slightly perturbed by other elements and by outside sources [54]. Examples of simple systems include oscillators and pendulum, whose behaviors are exactly predictable through the use of closed-form analytical equations.

Complicated systems can be defined as locally isolated systems: they can locally be decoupled into a set of weakly internally coupled one-component sub-systems that can be acted upon and controlled by external sources. A given system can be acted upon by as many sources as the number of elements in the system since each element can be distinguished from the others [54].

Complex systems can be defined as highly internally coupled systems whose dynamics are mainly determined by components' interactions. This is in agreement with the etymology: the word "complex" comes from the Latin "cum plexus" meaning "tied up with." Therefore, any action on such systems can only be performed globally and not at the component level, as was the case for complicated systems.

Complex systems differ from simple systems and from complicated systems through the concepts of self-organization and emergent behavior. Indeed, the components of complex systems are more sophisticated than those of simple or complicated systems. Components of complex systems are usually referred to as agents: these entities are searching through a collective behavior the satisfaction of properties they cannot reach individually. This feature of complex systems, called emergent behavior or feedback of "function" onto "structure," comes from the fact that components or agents are adapting their behavior from their interactions with other agents, and with the environment.

Finally, complex systems are less sensitive to outer control or action than complicated systems since their components are so strongly coupled that they cannot be identified: an internal control replaces a classical control from the outside. As a consequence, the more complex the system is internally, the less complicated it is when looked at from outside [53], [54]. Examples of complex systems include the human body, herds of animals, insect colonies, population activity in an economy, weather, and networks of computers.

To summarize, simple systems and complicated systems are usual ones nicely approachable by the methods of scientific reductionism [55], [56]. Complex systems, by their very global nature, cannot just be reduced to the effects of their components [57], [58]: they require some adjustment for being correctly handled because the key point is the way the system behaves under (or against) the action of its environment. In this context, “complex” and “complicated” are not synonyms: *complicated* tends to refer to large systems with many loosely coupled components, highly dependent on outside action, while *complexity* is derived from the internal interconnection and interaction of components of complex systems, that are mostly insensitive to outside control.

2.1.3. Detailed Definition of Complex Systems

As mentioned in previous chapters, the present study deals with the protection of critical assets and homeland borders, using fixed and mobile surveillance systems such as various types of radars and cameras. Needless to mention that the systems involved are complex systems, combined with one another, providing a capability to accomplish a given mission. Therefore, it is of interest to look more closely at the characteristics of complex systems to better understand their behavior and to maximize their ability to perform their intended mission. Collating ingredients of definitions from INCOSE [59], [60], University of Michigan [61], Clemson University [62], Mitre Corporation [63] and the New England Complex Systems Institute [64], the following properties of complex systems have been identified:

1. *Non-linearity*: complex systems exhibit non-linear dynamics and rarely any long-term equilibrium
2. *Interaction/Connection*: complex systems have many autonomous components (individual agents of the system), interacting or being connected in some fashion with each other
3. *Many nearly degenerate/equivalent configurations*: complex systems have many inherently heterogeneous components which can be arranged in a large number of potentially useful ways

4. *Hierarchies*: agents of complex systems are often organized in groups or hierarchies

5. *Emergent/self-organizing behavior*:
   - Complex systems display emergent macro-level behavior that results from the actions and interactions of the individual agents, but that cannot be predicted from them when studied in isolation
   - The structure and behavior of complex systems are not deductible, nor may they be inferred, from the structure and behavior of their component parts

6. *Adaptation and "Intelligent agents"*:
   - Complex systems adapt to their environment as they evolve: their complexity increases and the response of each of their agents changes according to the behavior of neighboring agents
   - Complex systems are non-deterministic: they exhibit an unpredictable behavior

7. *Fuzzy boundary*: the boundary of a complex system is often hard to define

8. *Multi-scale*: the structure of a complex system tends to highlight a number of different scales, any of which can affect the behavior of the system as a whole

Figure 4 shows a pictorial definition of a complex system with the above identified properties [64].

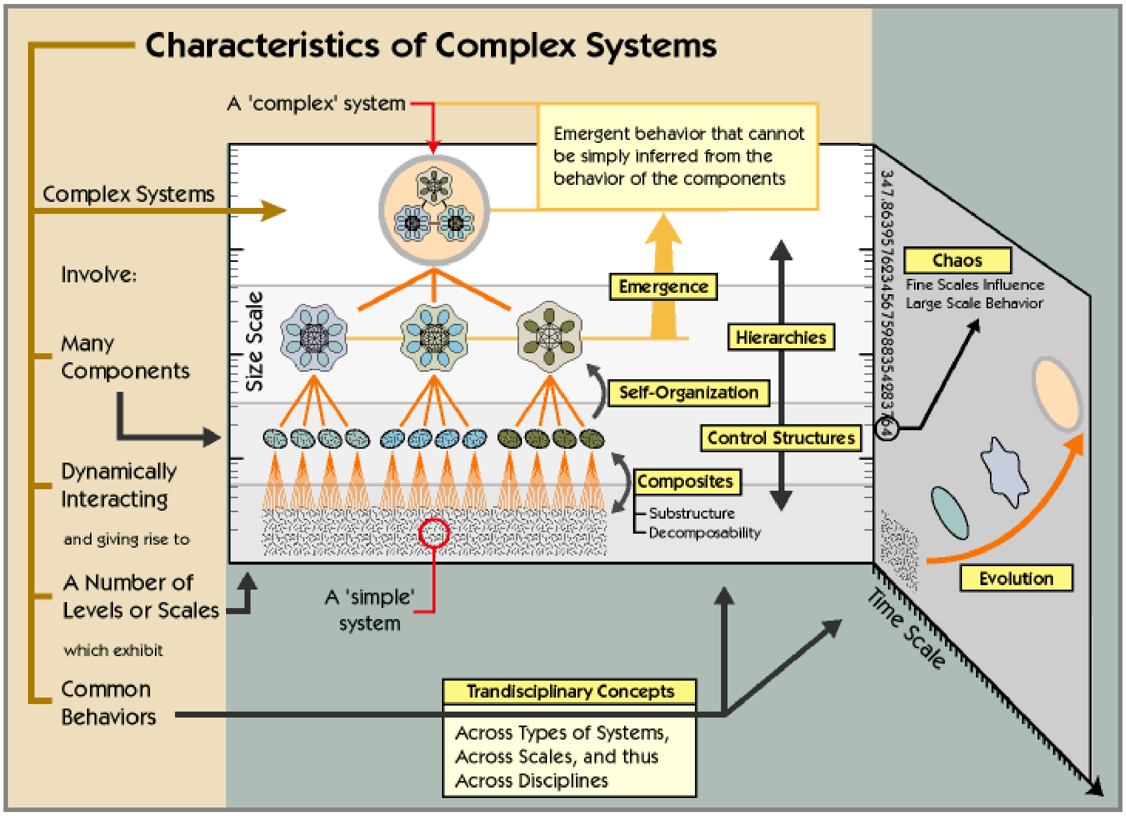


Figure 4: Complex Systems Representation

2.1.4. Systems-of-Systems

The next level of complexity in the field of systems study concerns the association of multiple systems to form a "system-of-systems."

2.1.4.1. <u>Definition</u>

The term "system-of-systems" has become an increasingly popular terminology for large scale systems composed of a variety of heterogeneous, interoperable, and collaborative systems. "*The term 'systems-of-systems' is generally used to define a class of systems wherein a set of independent systems, each having unique behavior and performance, is organized to perform collaboratively and coherently to achieve a* *purpose*" [65]. Therefore, there must exist a relationship among the system components, and a common objective or purpose for each component, "*above and beyond any individual objective or purpose for each component*" in accordance with a study from the Department of Defense [66].

Adding to the DoD principles, Maier [67] defines five principal characteristics describing a true system-of-systems (SoS), distinguishing it from complex systems:

1. *Operational independence of the elements*: the component systems of a system-of-systems can be usefully operated independently if the system-of-systems is decomposed
2. *Managerial independence of the elements*: the individual systems forming a system-of-systems are separately acquired, and therefore are actually run and maintained independently of each other
3. *Evolutionary development*: the development of a system-of-systems is evolutionary in that functions can be added, removed, and modified with experience in use of the system. Therefore, a system-of-systems never actually appears to be fully formed
4. *Emergent behavior*: the entire system-of-systems performs functions and carries out purposes that emerge from the actions and interactions of its component systems, but that cannot be localized to any of them in particular. The principal purpose of a system-of-systems is fulfilled by its emergent behavior
5. *Geographic distribution*: the independent component systems of a system-of-systems can only readily exchange information, and not substantial items like mass or energy

2.1.4.2. Challenges

Although Maier's criteria are usually used to identify a system as a system-of-systems, there exist, in the scientific community, some disagreements regarding the classification of systems as systems-of-systems, families of systems, federations of systems, architectures of systems, coalitions of systems, collaborative systems, interoperable systems, complex systems, complex adaptive systems, super-systems and so on. That is why the INCOSE [51] offers a new set of challenges unique to systems-of-systems:

1. *System elements operate independently*: each component system of a system-of-systems is likely to be operated by its own right, independently of the other and of the whole system-of-systems to which it belongs

2. *System elements have different life cycles*: the component systems of a system-of-systems may be at different stages of their life-cycle. While some are possibly in their development phase, others are already deployed and operational. In some cases, some component systems may be scheduled for disposal before the replacement systems are deployed.

3. *The initial requirements are likely to be ambiguous*: due to the different time scales characterizing a system-of-systems (see property 2 above), the requirements are likely to be ambiguous in the sense that they can be very explicit for already deployed component systems, while being very unclear or changing for component systems that are still in the design stage. As such, the requirements for a system-of-systems evolve as the component systems evolve and the design matures.

4. *Complexity is a major issue*: as new component systems are added to a system-of-systems, the complexity of the interactions involved increases rapidly and can hinder the definition of exchanges between component systems.

5. *Management can overshadow engineering*: the component systems of a system-of-systems can exist in their own right and can have different time scales essentially because they have their own product/project office. Therefore, the development of a system-of-systems is further complicated by the necessary coordination of requirements, schedules, interfaces, budget constraints and technology upgrades among the various entities involved.

6. *Fuzzy boundaries cause confusion*: unless a central entity is in charge of the definition and the control of the scope of a system-of-systems, and manages the interfaces and boundaries of the component systems, no one controls the definition of the external boundaries of the system-of-systems under consideration.
7. *System-of-systems engineering is never finished*: after all component systems of a system-of-systems are deployed and operational, management still needs to account for possible changes in the various component systems' life cycles, such as the emergence of new technologies or the necessity for replacement of component systems due to preplanned product improvement.

2.1.4.3. The Challenges of Complexity and Heterogeneity

According to the previous guidelines, the combination of multiple complex detection systems able to ensure the protection of critical assets and populations defines a system-of-systems, more commonly called architecture of systems. The primary difficulty with assessing detection capabilities is not the hierarchical nature of the problem. This could be the case for a more complicated study of the Design and Optimization of Detection Architectures (DODA) involving not only detection systems but also systems able to track and identify detected systems, and eventually systems for intercepting or addressing suspicious detected items of interest. Nevertheless, for the problem under consideration, the difficulty rather lies in the complex interactions between the components of the detection architecture. Indeed, such interactions can be temporary, evolutionary, or unpredictable, complicating the analysis of heterogeneous systems architectures.

Furthermore, what usually causes major troubles when designing multidisciplinary systems with complex interactions and many components is the evaluation of different options providing the same overall capability. Indeed, different detection architectures provide different levels of effectiveness and varying levels of cost, technology, and time for implementation and deployment. The complexity of such architectures is compounded by the fact that the detection systems of interest are rarely developed all at once. Indeed, there currently is a tendency to optimize the different elements of an architecture of systems in isolation, such that they are individually seen as ideal by the members of their respective design organizations. As a consequence, no structured methodology exists for the comparison of dissimilar systems against the same top-level measures of effectiveness.

Finally, the heterogeneity of complex architectures of systems adds to the somehow tedious task of organizing a system-of-systems into a hierarchy. While the distinction between a system-of-systems and a system can be debatable, the delineation between a system and a sub-system can be equally challenging. Indeed, depending on the point of view, an entity can be defined as either a system or a sub-system. For instance, to a car manufacturer, the car is viewed as a system and the engine is viewed as a sub-system. To an engine manufacturer, the engine itself is a system and is comprised of sub-systems such as the cylinder(s), the piston(s) and the valves. This challenge is summarized by Hatley [68] : "*every system below the level of the whole universe is a component of one or more larger systems. The larger systems are the context or environment in which the component system must work.*" An alternative way of addressing this challenge is to understand a system as the level at which you "are," a sub-system as the level at which you require to "function" and a system-of-systems as the level at which you "belong." This three-level scale depends on the perspective of the user. Therefore, there exist different types of descriptions of the same entity related to the different aspects of the entity being described. Each description represents a specific perspective, each standing alone, and each being different from the others, even though all the descriptions may pertain to the same entity and therefore are inextricably related [69].

2.2. <u>Systems Engineering and Systems-of-Systems Engineering</u>

As a result of the particularities and properties of complex systems and systems-of-systems highlighted in previous sections, a decomposition-based approach that does not take into account the strong interactions between system elements, is not appropriate for the thorough study of complex systems and systems-of-systems.

2.2.1. Systems Engineering

Although many disciplines have been developed for the analysis and design of different types of systems, the overarching scientific discipline for complex-systems study is called *Systems Engineering*.

According to the NASA Systems Engineering Handbook [70], "*Systems engineering is a robust approach to the design, creation, and operation of systems. In simple terms, the approach consists of identification and quantification of system goals, creation of alternative system design concepts, performance of design trades, selection and implementation of the best design, verification that the design is properly built and integrated, and post-implementation assessment of how well the system meets (or met) the goals. The approach is usually applied repeatedly and recursively, with several increases in the resolution of the system baselines (which contain requirements, design details, verification procedures and standards, cost and performance estimates, and so on).*" In the same vein, the Systems Engineering Fundamentals [71] defines systems engineering as "*an interdisciplinary engineering management process that evolves and verifies an integrated, life-cycle balanced set of system solutions that satisfy customer needs.*"

The International Council on Systems Engineering [60] further specifies: "*Systems Engineering is an interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design synthesis and system validation while considering the complete problem. Systems*

Engineering considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs."

Finally, the SMC Systems Engineering Primer & Handbook [53] introduces the notion of uncertainty in the design of complex systems through its definition of systems engineering: "*Systems engineering is a standardized, disciplined management process for development of system solutions that provide a constant approach to system development in an environment of change and uncertainty.*"

Therefore, in light of the above definitions, systems engineering is focused on the development of total systems solutions, including supportability, and operations and training, that satisfy the customer requirements while balancing cost, schedule, performance, and risk. As such, systems engineering is a process that is comprised of a number of activities that will assist in the definition of the requirements for a system, transform this set of requirements into a system through development efforts, provide for deployment of the system in an operational environment, and ensure the maintenance of the system throughout its life-cycle.

By its very nature, systems engineering is thus performed in concert with system management. Indeed, in order for the system manager to make the right decisions, the system engineer needs to provide useful and accurate information about the design, such as the identification and characterization of alternative design concepts. Also, an important aspect of the system engineer's role is to create models of the system so as to facilitate the assessment of the alternative concepts in various dimensions such as cost, performance, and risk.

The systems engineering definition of a system is particularly relevant to this research: "*an integrated composite of people, products, and processes that provide a capability to satisfy a stated need or objective*" [53]. Of greater interest though is to define the word "complex system" in the perspective of systems engineering. The following definition will be taken as a baseline for this research: *"Complex systems are systems that do not have a centralizing authority and are not designed from a known specification, but instead involve disparate stakeholders creating systems that are functional for other purposes and are only brought together in the complex system because the individual "agents" of the system see such cooperation as being beneficial for them"* [72].

2.2.2. Systems-of-Systems Engineering

As described in previous sections, a "system-of-systems" is inherently different from a "system." Therefore, it is safe to assume that there are principles of system-of-systems engineering that are different in essence from systems engineering.

Systems engineering has long been considered as the theory of everything so that there can be nothing that does not fall into its preview. Over ten years ago, Mark Maier's argument for denoting certain systems as "system-of-systems" was based on the fact that the principles applied differ between systems and system-of-systems. Using a similar argument, the difference in name and in essence between systems engineering and system-of-systems engineering is justified because of the principles that do apply to systems-of-systems but that do not hold for systems, especially the properties of emergent complex behavior, heterogeneity, and interaction. System-of-systems engineering is thus meant to stretch the boundaries of traditional systems engineering in three areas [73]:

- First, high levels of ambiguity and uncertainty pertaining to problems involving systems-of-systems are not addressed by traditional systems engineering. Systems-of-systems are characterized by highly dynamic and turbulent development and operational environments that most often than not result in shifts and pressures on problem definition and requirements. As a consequence, traditional systems engineering is unable to effectively and efficiently address problems with a high degree of contextual influence as is the case for a detection architecture.
- Second, contextual influences are usually placed in the background of the systems engineering process although the latter does not completely ignore the influence of the context on the system in the problem formulation, analysis, and resolution. Nevertheless, problems involving system-of-systems cannot be artificially separated from their context. Indeed, the circumstances within which such a problem is embedded can highly constrain and overshadow the technical analysis in determining system solution success. For instance, the topographic and climatic environments, along with the operational situation in which a detection architecture of distributed systems evolve, drive the physical and functional

design of the detection system-of-systems,. As such, they need to be fully incorporated in the definition and optimization of detection architectures.

- Third, systems engineering provides complete system solutions, optimally developed through iterative design and development processes. However, there is an increasing demand for deploying systems-of-systems that may offer incomplete or partial solutions that might be iterated upon after development, based on a range of resource, time or technological constraints. This may be achieved through the use of systems-of-systems engineering which provides a way to obtain fully or partially complete detection architecture solutions responding to specific surveillance and protection requirements. These detection architectures might then be completed or iterated upon to uncover solutions that best fit custom constraints.

2.2.2.1. Issues Relating to Systems-of-Systems Knowledge

Before detailing the concept of system-of-systems engineering, it is worth noting that there exist some discrepancies in current knowledge of systems-of-systems [73].

First, there is no universal definition for the term "system-of-systems."

Second, there usually is little development of the concept of system-of-systems beyond superficial explanations:

- There is no *"identified linkage/basis in [the] body of theory and knowledge"* pertaining to systems-of-systems
- There is an insufficient *"depth to demonstrate detailed grounding within one or more disciplines"* involved in systems-of-systems studies
- There is no *"empirical development of an associated body of knowledge for use as a guide, furthering knowledge development and practical applications"* [73]

Therefore, there exist no fundamental principles, underlying theory, accepted methodologies, or body of empirical work that would constitute foundations for a discipline pertaining to the engineering of systems-of-systems.

Third, system-of-systems concepts are mostly defined and applied in information technology, although the latter only offers a partial perspective for engineering a system-of-systems solution. This is due to the interdisciplinary nature of the systems engineering discipline.

Fourth, although systems engineering has been effective in providing methodologies, processes, and tools to cope with simple systems complexity, most attempts to deal with systems-of-systems using the traditional systems engineering process have been fruitless. Indeed, success of systems engineering at a single system level does not guarantee the success at the system-of-systems level.

Fifth, systems engineering mostly addresses the technical aspect of the problem being solved. Nevertheless, the contextual, human, organizational, policy, and political dimensions are just as important, and ultimately shape the decision space and the feasible solutions for the technical system problem.

2.2.2.2. Differences Between Systems Engineering and Systems-of-Systems Engineering

As evidenced in a previous section, there exist several definitions for systems engineering. In particular, Martin's definition illustrates a necessary shift in thinking about system-of-systems engineering: "*Systems engineering is the process that controls the technical system development effort with the goal of achieving an optimum balance of all system elements*" [74].

Systems engineering differs from system-of-systems engineering in many different ways. First, on the one hand, systems engineering only deals with single complex system problems. As such, the complete systems engineering process, including definition, analysis, and development, is meant to bring into being one complex system to address a specific problem or need. On the other hand, system-of-systems engineering focuses on the integration of multiple complex systems into a system-of-systems. The latter may involve already existing systems, newly designed systems, or a mixture of the two. The whole purpose of system-of-systems engineering is thus to form a system-of-systems comprised of possibly heterogeneous complex systems brought about by an emerging need or mission. Second, traditional systems engineering concentrates on system performance optimization [75]: "*Systems engineering is the management function that controls the total system development effort for the purpose of achieving an optimum balance of all system elements. It is a process that transforms an operational need into a description of system parameters and integrates those parameters to optimize the overall system effectiveness*." However, since complex systems are subjected to uncertainty, heterogeneity, complexity, and are constrained by resources, seeking solutions that work, also called "robust solutions", is actually preferable to searching for an optimal solution. System-of-systems engineering adopts this perspective [76]. The focus is now to develop satisfactory solutions to complex system problems such that they are appropriate for present and near future conditions and circumstances, bearing in mind that the deployed system solution may evolve based on demands that cannot be fully anticipated in advance of deployment. As such, system-of-systems engineering must remain flexible to adjust to shifting problem contexts and conditions. In particular, it has to primarily focus on the development of a methodology rather than a process. Indeed, when dealing with complex systems-of-systems, a simple n-step process is not enough: a more dynamic structure is of rigor in order to be compatible with the potentially rapidly changing conditions characteristic of system-of-systems engineering.

Third, as stated in the first point, systems engineering concentrates its efforts on the production of a system solution to address a problem or need. On the contrary, system-of-systems engineering focuses on the deployment of an initial response and not on a final system solution since the latter does not necessarily exists or is not necessarily expected.

Fourth, systems engineering has successfully been applied to clearly defined problems with relatively clear goals and boundaries. Nevertheless, such problems are not the majority when it comes to rapidly evolving complex systems-of-systems [77]. Therefore, system-of-systems engineering is based on a dynamic process able to cope with the increase in information intensity, contextual richness, and problem complexity in order to address emergent complex system-of-systems problems.

Fifth, as brought to attention earlier, while systems engineering reduces its concern to the technical aspect of a problem, system-of-systems engineering must increasingly appreciate contextual, human, and political influence in addition to the technical domain.

Sixth, systems engineering processes have been developed with the assumption that, once properly defined, the goals or objectives for a system are fixed and unitary, meaning that subsequent analysis could move forward and could always refer back to the systems goals if needed. This is no longer the case for complex systems-of-systems for which the objectives can be ill-defined, potentially pluralistic, shifting, and possibly ambiguous. For instance, two contractors working on the same project may have completely different tacit objectives which have nothing to do with the system performance goals, but which express their relative desires for profitability and power. That is why system-of-systems engineering is intended to consider and account for dynamic pluralistic system objectives.

Lastly, in systems engineering approaches, boundaries are identified primarily from a technical point of view and somewhat fixed throughout the analysis. For system-of-systems engineering however, boundaries are likely to be arbitrary, permeable and probably evolving throughout the analysis. In Mitroff's mind [78], inappropriately bounding a system problem is a major source of error for resolving complex system problems. Is this perspective, system-of-systems engineering seems to be at a deadlock. Nevertheless, in the context of complex systems-of-systems, boundary shift is usually not the result of a poorly defined problem, but rather a consequence of potential emergent technology, evolving requirements, or changing conditions. Therefore, system-of-systems engineering differs from systems engineering in that it has to be capable of recognizing and compensating for possible sudden or dramatic changes in system-of-systems boundaries.

Despite the aforementioned differences, system-of-systems engineering can draw on the following strengths of systems engineering to increase its ability to deal with more and more complex and dynamic system-of-systems problems [73]:

- "Linkage to systems theory and principles for design, analysis and execution"
- "Interdisciplinary focus in problem solving and system development"
- "Emphasis on disciplined and structured processes to achieve results"
- "Iterative approach to develop systems to meet expectations for problem resolution"

2.2.2.3. Perspectives of Systems-of-Systems Engineering

In "System of Systems Engineering" [73], the authors define system-of-systems engineering as "*The design, deployment, operation, and transformation of metasystems that must function as an integrated complex system to produce desirable results. These metasystems are themselves comprised of multiple autonomous embedded complex systems that can be diverse in technology, context, operation, geography, and conceptual frame.*" In this definition, the metasystem is the system-of-systems of interest, namely the distributed systems detection architecture, whose composition includes a hierarchy of multiple, possibly heterogeneous complex detection subsystems it is intended to integrate. These subsystems are within the boundaries of the system-of-systems. As such, their autonomy of decision, action, and interpretation is constrained by the interactions with other subsystems and the operation of the whole system-of-systems. Moreover, the detection subsystems are considered complex because they have the characteristics of complex systems enounced by Jackson [79]:

- Large number of variables or elements
- Rich interactions among elements
- Difficulty in identifying attributes and emergent properties
- Loosely organized interactions among elements
- Probabilistic (as opposed to deterministic) behavior in the system
- System evolution and emergence over time
- Purposeful pursuit of multiple goals by system entities or subsystems
- Possibility of behavioral influence or intervention in the system
- Largely open to the transport of energy, information, or resources across the system boundary from/to the environment

Finally, the complex detection subsystems are very much likely to be heterogeneous in multiple dimensions, namely *"technology, context, operation, geography, and conceptual frame"* [79], such that they need to be handled differently from a managerial point of view as well as from a conceptual point of view. Thus, there might be very diverse, even conflicting, worldviews driving perspectives within the system-of-systems, and responsible for the uniqueness of the constraints imposed on the system-of-systems by local circumstances and conditions of subsystems [80].

To summarize, system-of-systems engineering addresses the requirement for the development of more and more complex systems and their incorporation into increasingly integrated complex systems-of-systems. The systems-of-systems engineering approach is not necessarily an extrapolation of the traditional systems engineering process, and the successes of the latter in developing individual complex systems cannot be taken for granted when it comes to designing heterogeneous systems-of-systems. Systems-of-systems engineering is thus an evolution of systems engineering in the sense that it incorporates its strengths, its disciplined inquiry and rigor, while addressing its shortcomings in dealing with increasingly more complex system problems. In this context, systems-of-systems engineering is particularly appropriate for designing and optimizing distributed system architectures in the context of homeland security. Nevertheless, the accurate definition of detection systems-of-systems in specific operational environments requires the detailed analysis of each of its component. This way the physical and functional interfaces between the various elements of the systems-of-systems may be revealed and understood. This is performed through the use of parametric analysis where the detection architecture is first decomposed into its main components. Each component is then further analyzed so as to reveal a set of relevant attributes that facilitate the design, modeling, simulation, and optimization of the complete system-of-systems.

2.3. Parametric Analysis

As clearly stated in previous sections, the components of the detection system-of-systems have a certain level of autonomy: each element within the system architecture has its own distinct goals, such as self-preservation and maintenance of a certain level of safety [81], that it pursues in parallel with sub-goals defining its contribution to the achievement of the overall detection and surveillance goals. However, this independence property may have adverse effects on the system-of-systems in that it may entail conflicts of responsibility at the architectural level. Indeed, some partially or fully autonomous components of a system-of-systems may have such a degree of freedom that they become a risk to themselves and to the architecture as a whole [82]. Nevertheless, this characteristic of certain systems-of-systems consisting of more or less autonomous components, also works to their advantage. Indeed, it makes them very adaptable, thus allowing improvements to take place within different layers.

Moreover, the very nature of the detection system-of-systems, which, by definition, is composed of multiple, possibly heterogeneous components interacting with each other, makes possible the effective design of the system-of-systems by considering each of its components independently. Indeed, the detection architecture can be divided into smaller systems allowing for a simpler approach to be taken concerning the analysis of the system-of-systems' functional interfaces. This is first done by considering the detection system-of-systems as a whole, then decomposing it into its component systems, analyzing the components' interrelationships and interactions, and finally recomposing the detection architecture. A structured way to accomplish the decomposition of a system-of-systems into its components is to follow a functional decomposition technique, or more generally a morphological decomposition technique. Such decomposition consists of defining the elements of a system-of-systems from the high-level requirements they enable [83]. Originally, functional decomposition was developed by Pahl and Beitz [84], and Koller [85] to describe the direction of the flow information in a system.

In the case of the protection of critical assets and populations, the purpose is to decompose the problem in terms of the entities involved in an operational scenario. Such operational scenario takes place in a particular geographic environment, under specific climatic conditions and operational situation featuring items of interest that need to be detected. In such a scenario, detection systems may or may not already be deployed in the theater of operations. In any cases, the goal of such a scenario is to determine which detection architecture is best suited to provide adequate protection of certain assets or populations, i.e. which detection system-of-systems is capable of efficiently and effectively detect items of interest for the situation of interest.

In such a problem, the analyst is challenged both by the need to characterize a wide range of existing elements (such as various critical assets, various items of interest, and various detection systems), and the incentive to generate new notional ones. Hence, a structured and traceable, yet flexible, characterization scheme must be adopted. Such a scheme is built on the concept of *parametric representations*, whereby a set of parameters of interest with their respective domain of allowable values is identified and used to generate different configurations of a given element of the problem, or different elements altogether, depending on the application. To facilitate the generation of parametric representations, it is desirable to use a structured approach whereby the detection and surveillance problem is progressively decomposed, both physically and functionally, so that different elements can be adequately grouped, revealing sets of common parameters. A convenient way to structure the problem decomposition for parametric representations, and the subsequent synthesis, is through Morphological Analysis (MA).

2.3.1. Morphological Analysis

2.3.1.1. Origin and Applications

"... within the final and true world image everything is related to everything, and nothing can be discarded a priori as being unimportant."

- Fritz Zwicky

Discovery, Invention, Research through the Morphological Approach

Morphological Analysis (MA) was developed in 1942-1943 by the Swiss-American astrophysicist and aerospace scientist Fritz Zwicky, while at the California Institute of Technology (Caltech). It is a method for "*structuring and investigating the total set of relationships contained in multi-dimensional, non-quantifiable, problem complexes*" [86]. Zwicky applied it to a variety of problems such as the classification of astrophysical objects, the development of new forms of propulsive power systems (jet and rocket), and the legal aspects of space travel and colonization. He also founded the Society for Morphological Research and advanced the “morphological approach” for some 40 years, between the early 1930's until his death in 1974.

From the late 1960’s to the early 1990’s, a limited form of MA was employed by a number of engineers, operational researchers and policy analysts for structuring complex engineering problems, for developing scenarios and for studying security policy options. However, these earlier studies were carried out by hand or with only rudimentary computer support, which is highly time-consuming, prone to errors, and which severely limits the number and range of parameters that can be treated. That is why the Swedish Defence Research Agency later extended and computerized Morphological Analysis. As a result, the General Morphological Analysis (GMA) was developed in the middle of the 1990’s. The GMA is typically used for structuring complex policy and planning issues, developing scenario and strategy laboratories, and analyzing organizational and stakeholder structures.

Recently, Morphological Analysis and its derivatives have been successfully applied to a number of research topics in the U.S.A. and in Europe, in such various contexts as:

- Analyzing policy and futures studies
- Structuring complex policy and planning issues
- Relating means and ends in complex policy spaces
- Developing scenarios and scenario modeling laboratories, especially for military applications
- Developing strategy alternatives
- Analyzing risks
- Developing models for positional or stakeholder analysis
- Evaluating organizational structures for different tasks
- Presenting highly complex relationships in the form of comprehensible, visual models

Morphological Analysis has also been employed successfully for:

- Modeling society’s capacity to manage extraordinary events [87]

- Studying the protection of nuclear facilities against sabotage [88]
- Developing scenarios and strategies for the protection of nuclear facilities [89]
- Modeling multi-hazard disaster reduction strategies [90]
- Evaluating preparedness for accidents involving hazardous materials [91]
- Developing policies for the protection of the air transportation system [92]

To summarize, though MA is not new, it has certainly benefitted from recent advances in computation, and in turn has led to a number of successful methodological variants across a wealth of contemporary DODA applications, such as air transportation systems risk assessment [93], sabotage and attacks to nuclear power infrastructure, and other major events [94].

2.3.1.2. Method

Morphological Analysis is carried out by developing a discrete *parameter space* of the problem complex to be investigated, and defining relationships between the parameters on the basis of internal consistency. Such an internally linked parameter space is called a *morphological field*. The approach therefore uses a morphological matrix (sometimes referred to as a matrix of alternatives) to document how a system of interest is decomposed into main element classes, and enumerates possible alternatives for each element class. A Cross Consistency Matrix (CCM) then documents relational data between element alternatives, thus establishing the combinatorial logic that drives the synthesis of element alternatives into a number of internally consistent system configurations. The general method of Morphological Analysis is summarized in Figure 5:

1. In the first step, the problem under consideration is clearly and concisely stated so as to have a precise idea of what is at stake.

2. In the second step, the parameters of importance for the problem considered are identified. Alternatives are also defined for each main parameter, and ranges of values or conditions are brainstormed for each alternative.

3. In the third step, the information gathered in the previous step is regrouped in a variable and variable-condition matrix called a morphological field, which implicitly contains the solution space for the problem at hand. However, usually, the morphological field contains hundreds of thousands of theoretically possible combinations of alternatives/values/conditions.

4. Therefore, in the fourth step, the internal consistency of all pairs of variable conditions is assessed in order to weed out all inconsistent, contradictory or incompatible pairs. This step of the process is actually the most cumbersome and time consuming but also the most important. Indeed, more often than not, this is when it becomes clear that the variable conditions are poorly defined, i.e. that "we do not know what we are talking about." This leads to a review of the first two steps and iteration between steps 1 and 4 until the internal consistency assessment begins to work smoothly.

5. In the fifth step, an internally consistent outcome space is synthesized by going through all of the possible configurations in the morphological field and reducing the field by eliminating the combinations that contain internal contradictions. This is usually done with computer support. The outcome is the "solution space" of the defined problem.

6. Finally, in the last step, dependencies between alternatives of the "solution space" are analyzed, and adjustments to the variables and variable conditions are made as required.

It is worth noticing that all of the steps in a morphological analysis are iterative and therefore somewhat time consuming. Nevertheless, these iterations are necessary and valuable since knowledge about the problem develops and grows over time.

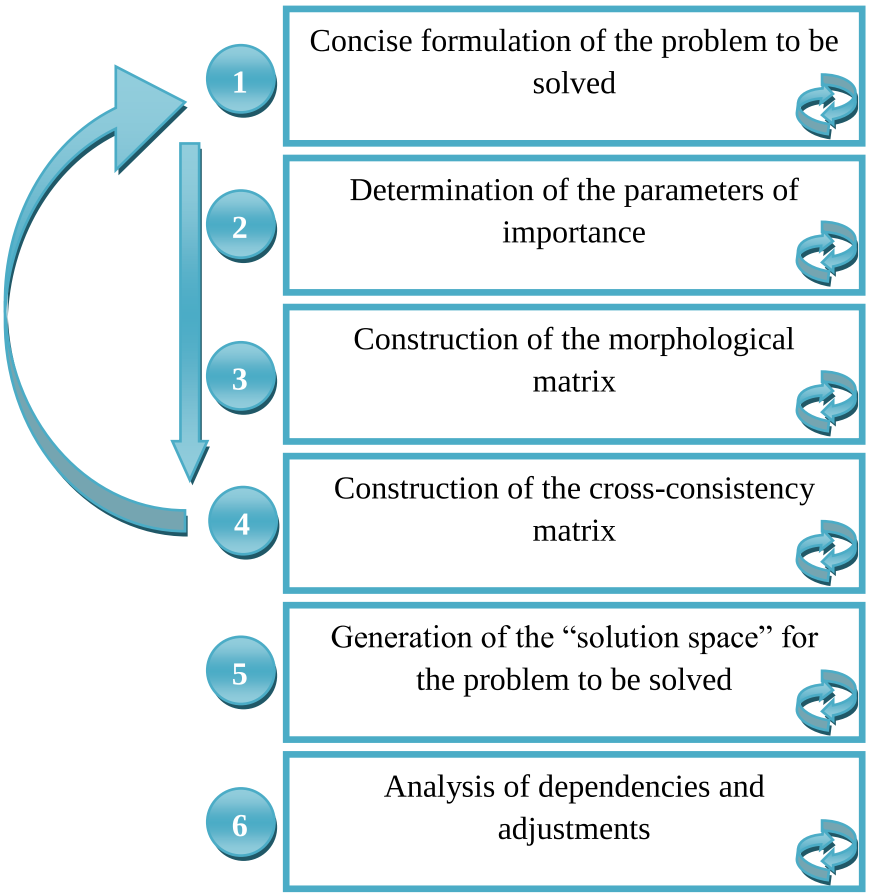


Figure 5: Sequence of steps for the Morphological Analysis (the blue circular arrows indicate iterative procedures within steps or between steps)

To summarize, the morphological approach has several advantages over less structured approaches in that it seeks to be integrative and to help discover new relationships or configurations. Additionally, it encourages the identification and investigation of the limits and extremes of different parameters within the problem space.

Nevertheless, these limits and extremes are usually based on ranges of values or alternatives that tend to be dictated by current state-of-the-art, although margins may be considered when delineating a problem space. The issue lies in the fact that future state-of-the-art in a particular field might not follow directly from current state-of-the-art in that same field. Similarly, current alternatives might become obsolete with time, while others might emerge that one could simply not have thought about at the time of the study. Therefore, there is an inherent uncertainty associated with the design of a complex system-of-systems in that, although one can try to forecast possible future states of the world, one can neither be hundred percent sure about which one will prevail nor if this particular one will ever be among the set of forecasted futures.

A NASA white paper describes those kinds of design problems that have a non deterministic formulation as *uncertainty-based designs* [95].

2.3.1.3. Morphological Analysis Applied to Critical Assets and Populations Protection

This research incorporates two important improvements to the traditional formulation of MA, as described in the previous section. First, the inherent hierarchical structure of the original MA formulation only provides two levels, that is, a single decomposition/synthesis step between a system and its elements. This research explicitly incorporates a multi-level approach accommodating any successive decomposition steps that may be required, thus more closely following the conceptual formulation of the systems engineering "Vee" [96]. This results in the creation of as many morphological matrices as that dictated by the multi-level decomposition approach. Then, the traditional MA formulation applies to a single complex system entity that is being analyzed. The present research expands the morphological analysis technique to the decomposition and synthesis of a complex problem, that of the surveillance and protection of critical assets and populations in specific operational situations. Such a problem may be considered as a system-of-systems containing various disparate elements that need to be assembled to generate an operational capability. Thirdly, traditional MA uses a binary scale in its cross-consistency matrix to determine whether two elements are compatible (1) or incompatible (0). The current research builds on previous efforts [93] to assess the consistency of the information contained in the morphological matrices, and encodes relational data with higher resolution scales to capture more complex interactions. More precisely, it implements a probabilistic scale instead of the traditional MA binary scale to introduce some nuance in the cross-consistency assessment. This once again results in the creation of as many cross-consistency matrices as that dictated by the multi-level decomposition approach.

With these improvements in mind, MA is used as a mechanism to structure and document the top-down physical and functional decomposition of the detection problem, as well as the relational data that drives its bottom-up synthesis back to the highest level of the hierarchical structure. The complete MA process as applied to DODA is described in Chapter 3.

2.3.2. Systems Engineering "Vee" Process

2.3.2.1. Definition

The Systems Engineering Process (SEP) is described by the DoD as an iterative process starting with requirements analysis, proceeding to functional analysis and requirements allocation, then to synthesis. The SEP is initiated by proper inputs from relevant customers, as indicated by the *"Process Input"* element on the left of Figure 6. Iteration can occur within each step of the core activities of Systems Engineering located inside the yellow oval in Figure 6, or via the requirements, design, and verification feedback loops. The outer feedback loop, labeled *"Verification Loop,"* ensures that the evolving design satisfies the functional and performance requirements, and meets the design constraints. The first inner feedback loop, labeled *"Requirements Loop"* checks that each function addresses at least one requirement and that, if this is not the case, the said function is either not needed, or a requirement is missing. Similarly, the second inner feedback loop, labeled *"Design Loop,"* ensures that each element of the design solution addresses at least one function. As for the *"Technical Management Processes,"* they represent a series of technical management activities listed on the upper right corner of Figure 6. These activities enable the selection and evaluation of alternatives, the assessment of progress, and the documentation of data and decisions throughout the project development. They can be separated into *"Decision Analysis"* activities and *"Technical Assessment"* activities as suggested by the red boxes on the upper right corner of Figure 6. Finally, the SEP ends with the generation of a *"Process Output"* such as a decision database, system/subsystem architectures, or specifications and baselines, depending on the level of application of the SEP and on the project.

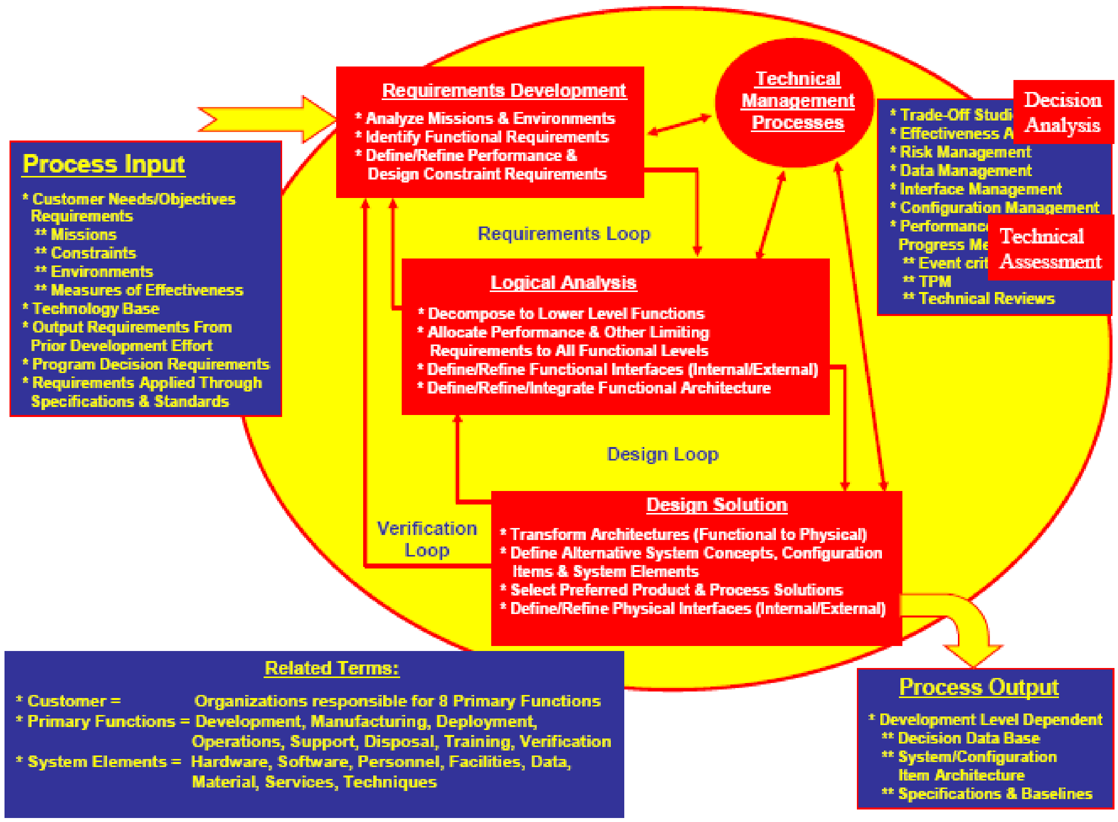


Figure 6: Systems Engineering in the DoD

2.3.2.2. <u>"Vee" Model</u>

The "Vee" model was developed by NASA as part of the Software Management and Assurance Program (SMAP) and modified by Forsberg and Mooz [96] in order to describe *"the technical aspect of the project cycle."* The "Vee"-shaped chart starts with user needs on the upper left and ends with a user validated system on the symmetric upper right. Figure 7 illustrates the example of a "Vee" chart in the context of Intelligent Transportation Systems [97].

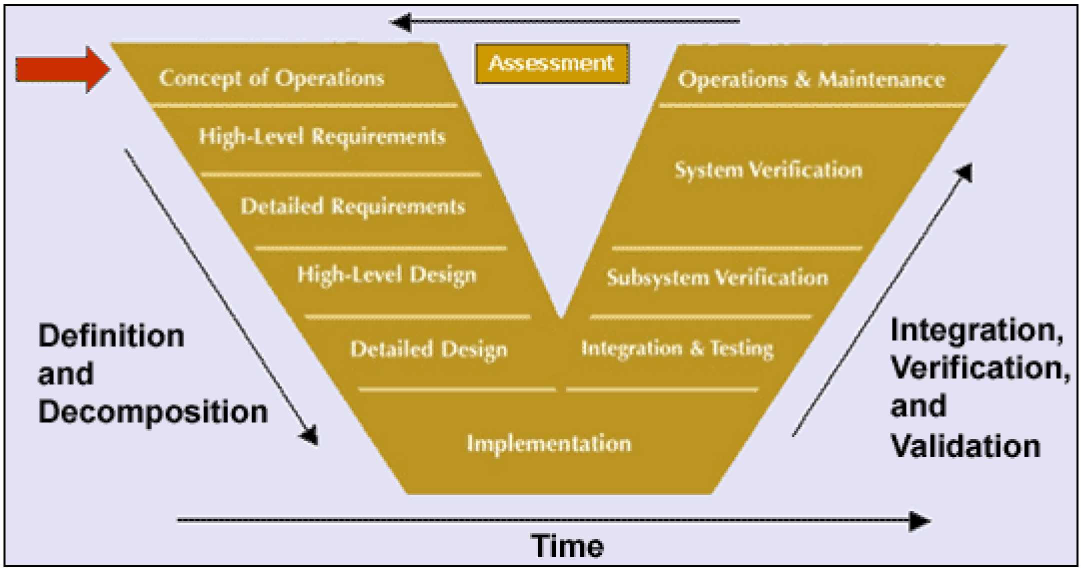


Figure 7: "Vee" Chart Example in the context of Intelligent Transportation Systems

As can be seen in Figure 7, on the left side of the chart, *"Definition and Decomposition"* flows down and to the right, as in a "waterfall" model. The "waterfall" model is a sequential software development model in which development is seen as flowing steadily downwards, like a waterfall, through the phases of requirements analysis, design, implementation, testing (validation), integration, and maintenance. The origin of the term "waterfall" is often cited to be an article published in 1970 by Winston W. Royce, although Royce did not use the term "waterfall" in this article. Ironically, Royce was presenting this model as an example of a flawed, non-working model [98]. The "waterfall" model is depicted in Figure 8 (it is Figure 2 in [98]).

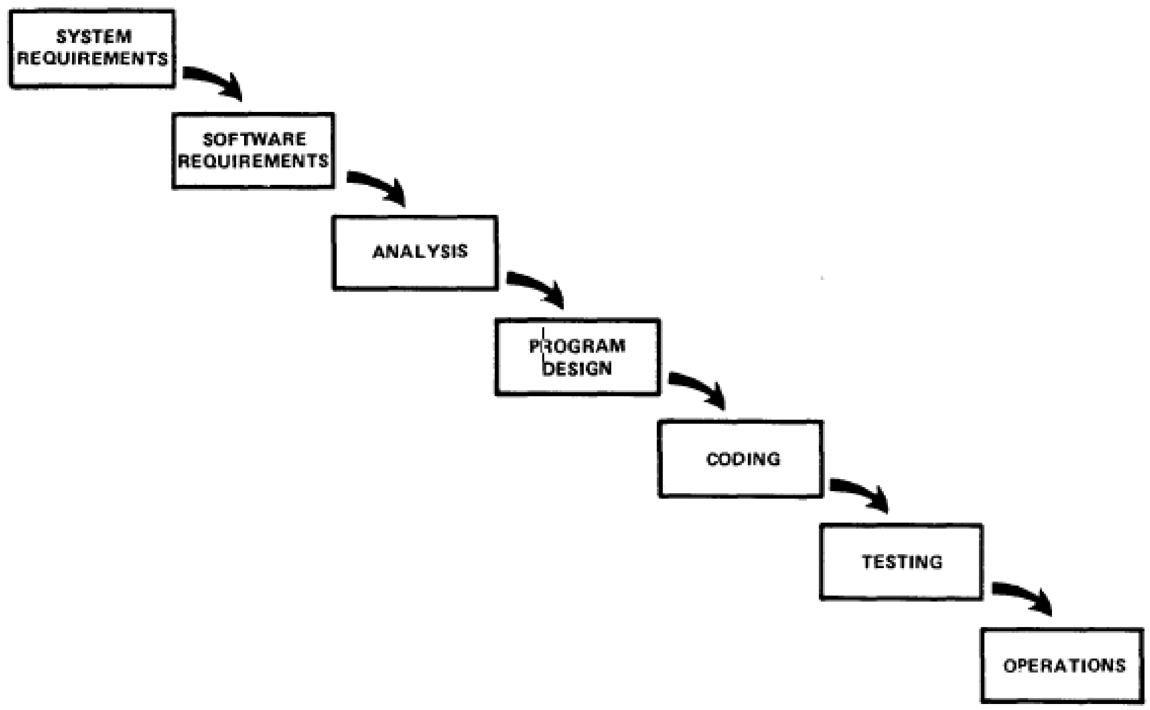


Figure 8: Original "Waterfall" Model as in [98]

In Figure 7, on the right side of the chart, *"Integration, Verification and Validation"* flows up and to the right as successively higher levels of assemblies, units, components, and subsystems are verified, culminating at the system level.

In other words, the early steps in the "Vee" chart define the project scope and determine the feasibility and acceptability as well as the costs and benefits of the project. Planning and programming/budgeting, which are intended to identify high-level risks, benefits, and costs, and to determine if the project is a good investment, are supported by these early steps. For their part, the last steps are intended to support project implementation, then transition into operations and maintenance, changes and upgrades, and ultimate retirement or replacement of the system when appropriate.

2.3.2.3. Definition and Decomposition

The "Vee" chart typically provides a three-dimensional view of a project. Indeed, though not shown in Figure 7, each box in the "Vee" represents a number of parallel boxes symbolizing subsystems that may make up the system at the level of decomposition considered. For instance, in the case of the top left box, labeled *"Concept of Operations,"* the various parallel boxes would represent alternative design concepts being evaluated in the early stages of the project. As the project progresses, a baseline is established for each alternative design and is put under formal configuration management as soon as it is approved. This ensures that that the requirements do not change once the baseline design has been accepted.

As mentioned earlier, the left side of the "Vee" chart is similar to the "waterfall" model or "requirements-driven design" model of a project development process. This side of the "Vee" represents the evolution of user requirements into system, subsystem, and component level requirements and eventually into the system design solution through the process of decomposition and definition.

Besides, in accordance with the "requirements-driven model," the project is punctuated by control gates defining significant decision points, beyond which work should not progress until documents containing the agreed upon decisions have been published and controlled by the project manager. However, this necessary precaution does not exclude conducting detailed work early in the project. In fact, detailed analysis and design in the very earliest stages of the project may be beneficial. Such an approach can help clarify user needs or establish credibility for the claim of feasibility of the design. Furthermore, implementation of *concurrent engineering* early in the project implies that off-core processes are included into systems engineering activities at each level of the "Vee." Such off-core processes involve system design, advanced technology development, trade studies, risk management, and specialty engineering analysis and modeling. These activities are performed at each level of the "Vee" and may be repeated multiple times within a phase, possibly leading to many kinds of studies and decisions. Nevertheless, only decisions made at the core level are put under configuration management at the various control gates. For their part, off-core activities, analyses, and models are not formally controlled but rather are used to complete and support the core decisions and to ensure that risks have been determined to be acceptable or mitigated. Therefore, analyses, data and results associated to off-core work still needs to be archived to facilitate replication at the appropriate times and levels to support introduction into the baseline. As a consequence, there should be enough iteration downward in order to establish feasibility and to identify and quantify risks. However, upward iteration with the requirements statements should be kept to a minimum. The user is allowed to generate new requirements or to change existing ones up or down to a certain level and until a certain time into the project. Indeed, upward confirmation of solutions with the users is sometimes necessary when user requirements cannot be adequately or concisely formulated at the inception of the project. Nevertheless, if upward iterations are maintained for too long, cost and schedule are likely to get out of control. Past a certain point, called the "Project Design Review" (PDR), modification of user requirements should be held for the next model or the next release of the project. If some modifications are to be made after the PDR, then the project should be stopped and restarted with a new "Vee" chart, completely reinitializing the project. In such a case, restarting the project may be more appropriate given the lessons learned but all the steps have to be redone from the start.

Finally, time and project maturity flow from the upper left to the lower right on the *"Definition and Decomposition"* side of the "Vee" chart. Therefore, once a control gate is passed, backward iteration is not possible.

2.3.2.4. Integration, Verification and Validation

The right side of the "Vee" chart is devoted to the *"Integration, Verification and Validation"* process. This side of the "Vee" represents the integration and qualification of the final design solution. The integration and qualification of the system can be further broken down into the lower level processes of integration, test, delivery, and fielding of the system. In this *"Integration, Verification and Validation"* process, activities are ascending from left to right and are in direct correspondence with activities of the *"Definition and Decomposition"* process which are descending on the left side of the "Vee." This is obviously voluntary in the sense that the verification method must be determined as the requirements are developed and documented at each level of the project. This ensures that requirements are formulated in such a way that they can be measured and verified. This is why the system verification approach must be determined at the highest level, as user requirements are translated into system requirements. Besides, cost and schedule can be driven by the technical demands of the verification process, which can further lead to a choice between alternative concepts. For instance, engineering models that are to be used for verification and validation purposes must be specified and studied in terms of their characteristics, their cost, and their development time which has to be introduced into the project schedule from the start.

Moreover, if it happens that the user requirements are not well-specified or are too vague to permit final approval at the PDR, the project can be developed incrementally. The first step would be to make sure that a minimum set of user requirements is met. Then, subsequent steps would try to provide additional functionalities and performance based on user feedback, i.e. new or modified requirements. This incremental approach is necessary when the users are not aware of what they are really looking for right from the beginning or when the project requires several iterations with the users in order to progress in the right direction. This is relatively easy to describe in terms of the "Vee" chart: all increments have a common heritage which is the PDR and each "release" or "iteration" has its own "Vee." Therefore, the project development process is characterized by a series of displaced and overlapping "Vees" corresponding to the several "iterations" with the users.

2.3.2.5. Concurrent Engineering

As defined by NASA [70], "*concurrent engineering is the simultaneous consideration of product and process downstream requirements by multidisciplinary teams.*" In this scheme, specialty engineers from all disciplines involved in the project must be part of the project team at each stage of the development process, so that they can provide their expertise when necessary during the system life cycle. Therefore, the system engineer is responsible for ensuring that all disciplines are represented in the early phases of the project, even if they happen to effectively come into play in later stages of the design process. This ensures that accepted requirements can be met and that selected design concepts can be built, tested, operated and/or maintained. Indeed, if technical experts are not included early enough in the design process, then the project is more likely to pass early control gates prematurely, resulting in a need for significant iteration of requirements and designs later in the development process. This, in turn, can result in increased costs and shifted schedules.

2.4. Modeling and Simulation

"*The purpose of computing is insight, not numbers*"

- Richard Hamming

Modeling, defined as "*the representation, often mathematical, of a process, concept, or operation of a system, often implemented by a computer program*," is the act of creating a model of a system, process, or observed phenomenon [127]. It is often associated with simulation, defined as "*the representation of the behavior or characteristics of one system through the use of another system, especially a computer program designed for the purpose*" [127].

Furthermore, according to the terminology established by the Department of Defense, a model is a "*physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process*," and a simulation is a "*method of implementing a model over time. Also, a technique for testing, analysis, or training in which real-world and conceptual systems are reproduced by a model*."

According to Dieter, "*a model is an idealization of part of the real world that aids in the analysis of a problem [...] Simulation is a manipulation of the model*" [107]. The manipulation of the model consists of sequentially varying its input conditions and observing its behavior. Generally, a model can be described as being either descriptive or predictive, in the sense that it can either enable understanding a real-world system, or enable understanding and predicting its performance.

Dieter further classifies models as being either [128]:

- Static or dynamic
- Deterministic or probabilistic
- Iconic, analog or symbolic

The first level of classification deals with the time dependence property of the model: the model is said to be static if its properties are independent of time while it is said to be dynamic if its properties vary with time. The second level of classification deals with the uncertainty associated with the prediction of the outcome of an event by the model: the model is said to be deterministic if the outcome is known with certainty while it is said to be probabilistic if the outcome is not known with certainty. The third and last level of classification deals with the nature of the model itself. The model is said to be iconic if it physically represents the entity being modeled: this is the case of a wind tunnel model of an aircraft wing. The model is said to be analog if it only represents the behavior of the entity being modeled without physically describing it: this is the case of a flow chart. The model is said to be symbolic if it only represents abstractions and/or quantifiable components of the physical entity being modeled: this is the case of mathematical equations. Therefore, for example, a physical representation of an entity that can be described by random or probabilistic variables that can vary with time would be a dynamic-probabilistic-iconic model.

Finally, according to Przemieniecki [129], symbolic models can be further broken down into two categories, namely analytic models or simulation models. The analytic model yields exact solutions to deterministic outcomes, whereas the simulation model yields converged solutions to very complex problems characterized by uncertainty and possibly risk [130]. Przemieniecki then differentiates a deterministic simulation model from a probabilistic simulation model.

To summarize, the aforementioned model classes can be hierarchically related, as shown in Figure 9.

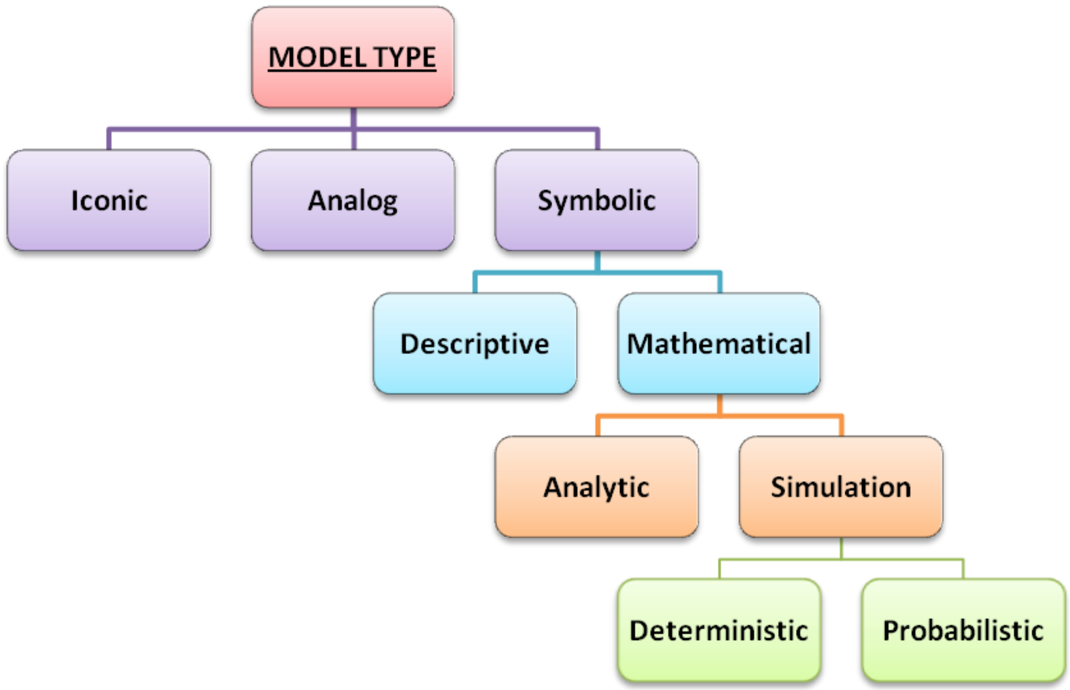


Figure 9: Model Classification [129]

In the case of the surveillance and protection of critical assets and populations, the elements of the problem may be modeled both physically and symbolically and described by probabilistic variables that can vary with time. These elements may also be simulated to study their performance under operational conditions so that customized detection architecture solutions may be uncovered. Therefore, the problem of DODA under study may be encapsulated in various types of models: dynamic-symbolic-mathematical-simulation-deterministic models will be used to model the behavior of detection systems, dynamic-symbolic-mathematical-simulation-probabilistic models will be used to model the behavior of items of interest, while symbolic-mathematical-simulation-probabilistic models will be used to model weather conditions and symbolic-descriptive models will be used to model the critical asset under consideration.

Depending on the problem at hand, a particular type of model of the system, entity, phenomenon or process of interest might be more suitable than another. There exist different ways of representing a system, entity, phenomenon or process, depending on the context of study, and on the level of insight expected from the modeling and simulation environment. However, there is no such thing as the best model: the choice of a model depends on several factors such as the goal of the model (for instance knowing the detailed physics), the modeler's background (statistics, engineering or physics), and the time available for developing the model. The main properties of the most commonly used design models in the field of engineering are exposed subsequently.

2.4.1. Stochastic Design Model

This is the most realistic model of all kinds. It is a function of both time and a set of variables describing the system, entity, phenomenon or process under consideration. Among those variables, at least one is random, i.e. is characterized by a probability distribution. Therefore, a stochastic model is both dynamic and probabilistic. Moreover, both the ranges of the design variables and the uncertainty vary with time. Finally, the goal of a stochastic model is to capture the evolution of the design over time.

2.4.2. Probabilistic Design Model

Contrary to a stochastic model, a probabilistic model is only a function of a set of variables describing the system, entity, phenomenon or process under consideration. Among those variables, at least one is random, i.e. is characterized by a probability distribution. Therefore, a probabilistic model can be considered a snapshot in time of the evolution of a stochastic model, i.e. a limiting case of a stochastic model at a fixed instant in time. In this case, the outcome of an event is not known with certainty.

2.4.3. Deterministic Design Model

A deterministic model is a function of a set of non-random or deterministic variables describing the system, entity, phenomenon or process under consideration, and possibly a function of time. Therefore, a deterministic model can be considered a snapshot of a probabilistic model in the probability space, i.e. a limiting case of a probabilistic model as the variance of the random variables approaches zero. In this case, the outcome of an event is known with certainty. A physics-based model is sometimes called a deterministic model although this is arguable. Indeed, a physics-based model follows from fundamental physical laws such as conservation of mass or Newton's laws of motion. A physics-based model involves measurable parameters and provides precise and trustable insight into the physical process that it models. It is therefore more mechanistic than deterministic in nature.

2.4.4. Virtual Design Model

A virtual model is the assessment of real-time interactive computer simulation of physical interactions in engineering systems.

2.4.5. Types of Design Models Involved in the Design and Optimization of Detection Architectures

In order to define detection architectures responding to specific surveillance and protection constraints in the context of homeland security, several types of models are implemented.

- Physics-based models are used to model the behavior of detection systems
- Physics-based probabilistic models are used to model the behavior of items of interest
- Probabilistic models are used to model weather conditions

- Simple descriptive models are used to model the geographic environment

All of the above models are finally implemented in a stochastic framework.

2.5. <u>Optimization</u>

Whether it be in physics, biology, economy or sociology, researchers often have to deal with the classical problem of optimization. In the majority of cases, purely analytical methods widely prove their efficiency. Nevertheless, they suffer from an insurmountable weakness residing in the fact that reality rarely obeys to the wonderful differentiable functions on which standard analytical optimization methods are based.

With the advent of new research areas such as that of system-of-systems and distributed sensor networks from which the problem under consideration is extracted, analytical methods proved less and less effective at finding an accurate solution to the optimization problem. Indeed, traditional optimization often fails in the above studies because requirements are either not well known at project start, or creep as a project matures, or include subjective criteria which are not easily accounted for. Therefore, several other optimization methods, combining mathematical analysis and random search have been developed. In such methods, one can imagine small robots searching a mountainous landscape representing the function to be optimized and wandering around to find the highest mountain. The robots can follow the path of steepest ascent they find. When one of them reaches a mountain top, it can claim that it has found an optimum. The method is very efficient if the mountainous landscape is composed of a single mountain, but in the opposite case, when the landscape contains several peaks of different altitudes, then there is no proof that the highest peak or optimum has been found: each robot could be blocked at a local optimum. Such kinds of methods actually only work with reduced search spaces. Systems-of-systems and distributed sensor network studies almost always involve multimodal functions, such that several local optima can be scattered around a global optimum. Hence, the challenges reside in the necessity of optimizing non-smooth data as efficiently as smooth data, and searching for global optima as effectively as local optima.

The problem of DODA under study implies the optimization of a detection architecture (system-of-systems) in terms of the types, the number, the properties and the positions of component detection systems. This has to be performed under the constraints imposed by an operational scenario, featuring various items of interest evolving in a particular geographic and climatic environment. DODA therefore appears to be a high dimensional, non-linear, discontinuous, and multi-criteria optimization problem. In this context, several optimization methods have been identified, that could prove potentially applicable to the optimization of detection architectures for the protection of critical assets and populations as part of the more global topic of distributed sensor networks. The main characteristics of each method are described subsequently, and a comparison of the most promising optimization techniques for the DODA problem is provided as a conclusion.

2.5.1. Distributed Sensor Networks

Surveillance of large terrains and geographic borders using a limited detection capability has always been an issue in the defense and protection community of many developed countries. Geographic borders are usually very long and may go through isolated areas with little or no inhabitants. As a result, it is almost always impossible to completely secure each and every mile of border against potential smugglers, drug dealers, illegal border crossing, and malevolent entities who are trying to enter a country to exploit or harm its people or infrastructures. In addition, the costs associated with the development of efficient architectures of protection systems may be so large that they become prohibitive. A compromise must therefore be made and a balance must be found between the level of acceptable protection and the cost governments are willing to invest. However, different countries may have different preferences when it comes to compromising between protection efficiency and cost, mostly depending on their histories and related experiences. Capturing such disparities is not straightforward and often requires the use of a parametric analysis or cost/benefit analysis. In addition, different preferences result in different protective solutions. Therefore, there exists a need for a method able to provide customized protection alternatives for varying levels of performance and cost. Such a method is found in the domain of optimization which enables the design of protection architectures that perform efficiently in a wide range of external conditions. These architectures are defined as Pareto optimal and are obtained by performing parametric optimizations.

Although optimizing sensor location over large terrains is a crucial need for the defense and protection community, very little work has been published in this domain in the open literature. For instance, in 1994, Franklin et al. [131] worked on optimizing the number and placement of observers on 1D and 2D terrains so that each point of the terrain could be visible from at least one observer. To do so, they proposed a greedy algorithm where the terrain elevation is given at the vertices of regular grids. However, this approach does not work properly on terrains having multiple local optima where sensor systems can be placed. Then, in 1998, Kewley and Embrecht [132] studied the positioning of military combat units for optimum performance. Contrary to Franklin et al. [131], their goal was not to entirely cover the terrain but rather to successfully complete a tactical mission in a battle. For this, they developed a fuzzy genetic algorithm in which potential solutions are evaluated through the simulation of a battle and a fuzzy logic module maps the simulation results to a single fitness value that is optimized by the genetic algorithm. In 2000, Can [133] created a meta-heuristics based on genetic algorithm to study the placement of sensor platform systems on large terrains. Then, in 2001, Vasquez and Hao [134] developed a tabu search algorithm to study the positioning of antennae for radio network planning. In a first step, they applied a constraint-based preprocessing approach to determine a set of efficient positioning sites offering the best performance given a set of predefined sites. In a second step, they used tabu search to optimize the number, types, and design parameters of the antennae in the network. In a third and final step, they developed a post optimization algorithm to improve the solutions given by the tabu search. At the same time, Kim and Clarke [135] developed new spatial optimization techniques to search for the locations which offered the best view of a topological surface in Geographic Information Systems (GIS). They investigated four algorithms - an extensive iterative search technique which evaluates each grid location on the surface, a conventional Tornqvist-based spatial search algorithm, and two evolutionary optimization algorithms, a genetic algorithm and a simulated annealing algorithm – and compared their ability to solve a visibility site selection problem. They showed that even with poor starting configurations, both the genetic algorithm and the simulated annealing can generate acceptable solutions. In 2003, Dhillon et al. [136] studied the problem of optimizing the coverage of distributed sensor networks for sensor resource management in the military domain, under the constraints of imprecise detections and terrain properties. For this, they developed a polynomial-time sensor placement optimization algorithm in order to determine the minimum number of sensors required and their locations so as to satisfy a given level of terrain coverage and preferential coverage of predefined terrain sites. More recently, in 2007, Murray et al. [137] worked on identifying an optimal configuration of video surveillance systems and their locations in order to maximize the coverage of large 3D urban areas to support security monitoring. Starting from a 3D terrain model, they modeled the video cameras and their locations by the Maximal Covering Location Problem (MCLP) and the Backup Coverage Location Problem (BCLP). Then, they combined a coverage optimization algorithm with a visibility analysis to determine the optimal configuration and placement of the surveillance systems improving both coverage and efficient resource allocation. In 2008, Krause et al. [138] looked into the problem of optimizing sensor placement to detect malicious introduction of contaminants in large water distribution networks. They investigated a two sided approach in which they combined either a greedy algorithm or a Mixed Integer Programming approach with a local search to improve the final solution. They showed that their approach naturally extends to multi-criteria optimization problems with large numbers of variables by analyzing the trade-offs between various objective functions and computing Pareto frontiers. At the same time, Bottino and Laurentini [139] developed an incremental Edge Covering (EC) sensor placement technique to solve the problem of positioning a minimum number of visual sensors in 2D able to cover the edges of a specific polygon. They combined an Integer Edge Covering Algorithm (IECA), a Lower Bound Algorithm (LBA) and the Indivisible Edges incremental Algorithm (INDIVA) to balance computational times and closeness to optimality even for problems of large dimensionality. In 2009, Hsieh et al. [140] developed a two-phase approach to determine the optimal set of surveillance cameras to secure a specific asset. First, they applied the immune-based algorithm to obtain the number, types, and locations of surveillance cameras in the optimal solution. Then, they further improved the locations of the surveillance cameras by assuming that the maximum failure detection probability was the same at all points of the asset. This two-phase approach significantly improves the final solutions while covering a wide range of budgets. In addition, they showed that as the budget increases, the final surveillance system is larger and composed of more efficient cameras. This decreases the failure detection probability until a large enough budget is reached, after which the failure detection probability tends to an asymptotic value. In 2010, Hamel et al. [141] simulated and optimized sensors to detect and track the release of airborne toxins in a large urban environment. They integrated models from Computational Fluid Dynamics (CFD), an integer-programming-based technique, and population mobility dynamics to determine the optimum sensor configuration able to mitigate the effects of released contaminants on civilian populations by providing accurate prediction of agent dispersion depending on the season and on the prevailing winds. The same year, Xu et al. [142] used a Particle Swarm Optimization (PSO) to improve the coverage of a large area by a network of cameras with known locations and type parameters. The problem of optimization consisted in determining the optimal orientation of each camera by either minimizing the number of cameras in the network to monitor a fixed area, or maximizing the coverage of a network composed of a fixed number of cameras. Constraints such as regions of variable importance and possible obstacles in the field of view of the cameras can be taken into account to cover a wider range of 3D applications. Finally, in 2012, Tao et al. [143] explored a polynomial-time algorithm to optimize the barrier coverage in directional sensor networks. The problem consisted in determining the minimum number of sensors and their orientations so that they can efficiently detect events occurring along the barrier joining the sensors.

Although examples of sensor placement in the domain of defense and protection do not abound, a rather extensive literature can be found in its encompassing area of distributed sensor networks [144],[145],[146],[147],[148],[149],[136],[150],[151]. In recent years, distributed sensor networks have been the focus of interest in numerous applications such as border protection [139],[143], surveillance and security monitoring [140],[142],[137],[152],[136],[153],[154],[141], water distribution and water quality monitoring [138],[155],[156],[157],[158], pollution monitoring [159], damage detection and characterization [160],[161], fault detection and diagnosis [162],[163], structural health monitoring [164],[165],[166],[167],[168], etc [150]. In these studies, various types of optimization algorithms were implemented, such as Glowworm Swarm Optimization (GSO) [144], Ant Colony Optimization (ACO) [145], Particle Swarm Optimization [160],[169],[142],[146],[170],various types of Genetic Algorithms [171],[172],[168],[161],[162],[164],[155],[165],[156],[173],[166], Simulated Annealing [170], Multi-objective Optimization [157],[158],[136], Integer Linear Programming [148],[163], combination of the NeuroEvolution of Augmenting Topologies (NEAT) method [174] and Pareto-dominance to create the Flexible algorithm for sensor placement FLEX [149], pareto optimization [159], polynomial-time algorithms [136],[143], stochastic optimization [175], and many others [167],[176],[150].

Coverage is a fundamental and widely accepted metric to evaluate the performance of surveillance systems in domains such as intrusion detection and border surveillance. The major goal in such applications is to detect intruders as they penetrate a protected region or before they cross the border. This constitutes a barrier coverage problem. Unlike other applications, the above problems do not require the coverage of each and every point of a given region but rather the coverage of that portion of the region through which intruders can penetrate. As such, the sensors can be deployed as a barrier to decrease the cost and achieve an acceptable coverage. However, coverage and cost are conflicting objectives in the sensor placement problem since increasing the number of sensors in the network increases both the coverage and the cost although what is looked for is to get the maximum coverage at the minimum cost. Multiple conflicting objectives call for multi-objective evolutionary optimization algorithms such as Genetic Algorithm and Particle Swarm Optimization which have proven to be well suited to tackle the problem of sensor placement under a variety of situations, in particular for border surveillance and intrusion detection [150],[151],[143],[140],[177],[178].

2.5.1.1. Linear Programming (LP)

Linear programming, also called linear optimization, is a specific case of mathematical programming or mathematical optimization. Is has first been developed secretly by Kantorovich in 1939 to reduce costs to the army during World War II by planning expenditures and returns [179]. In 1947, Dantzig unveiled linear programming to the public by publishing the simplex method [180] and Von Newman developed the theory of duality to solve linear optimization problems [181]. After the war, linear programming was essentially used in the fields of daily planning and game theory.

Linear programming is a technique used to determine the maximum profit or the lowest cost of a given linear, real-valued, affine objective function, subject to linear equality and linear inequality constraints or requirements [182]. In this context, the feasible space is a convex polyhedron, obtained by the intersection of a finite number of half spaces defined by the linear inequality constraints. The goal of the linear programming algorithm is then to find a point inside or at the nodes of the polyhedron where the function has the smallest or the largest value if such a point exists.

Linear programming problems can be defined canonically as follows:

$$\text{maximize or minimize } x_0 = c^T x = \sum_{i=1}^{n} c_i x_i$$

$$\text{subject to } Ax \leq b \text{ or } \sum_{i=1}^{n} a_{ij} x_i \leq b_j \text{ for } j=1,\ldots,m$$

$$\text{with } x_i \geq 0 \text{ for } i=1,\ldots,n$$

Where *x* is the unknown vector of positive variables to be optimized, *c* and *b* are vectors of known positive coefficients, *A* is a known matrix of positive coefficients, and x_0 is the objective function. The inequality constraints specify the convex polytope over which the objective function is to be maximized or minimized. The above standard form of linear programming problems is therefore composed of three parts [183]:

- A linear objective function
- Non-negative variables
- Problem constraints

Before a linear programming problem can be solved by the simplex algorithm [184], it is necessary to introduce non-negative variables, called slack variables, in order to transform inequality constraints into equality constraints [185]. In this case, the linear programming problem can be written in a block matrix form as follows:

maximize Z

satisfying:

$$\begin{bmatrix} 0 & -c^T & 0 \\ 1 & A & I \end{bmatrix} \begin{bmatrix} Z \\ x \\ x_s \end{bmatrix} = \begin{bmatrix} 0 \\ b \end{bmatrix}$$

Where Z is the variable to be maximized, $x \geq 0$, and $x_s \geq 0$ is the newly introduced slack variables. The above linear programming problem is said to be in an augmented or slack form.

Linear programming can be applied to various fields of study such as business [230],[231], economics [232], engineering [148],[163], transportation [233],[234], energy [235],[236], telecommunications [237], manufacturing [238],[239], and operations research. Indeed, several problems can be expressed in a linear programming formulation, including network flow problems [241],[242] such as planning, routing, scheduling, assigning, and designing, microeconomics and company management issues [243], such as planning, production, transportation, and technology, but also maximization of profits or minimization of costs with limited resources in modern management problems.

More details about duality theory, covering and packing problems, solution space, and methods for solving linear programming problems may be found in Appendix A.

2.5.1.2. Stochastic Optimization

Stochastic optimization methods are to stochastic problems what deterministic optimization methods are to deterministic problems. Stochastic problems are generally formulated as a function of random variables, and may involve random objective functions, or random constraints. Stochastic optimization methods therefore generate random iterates to solve stochastic problems [244]. In some cases, the function values may be contaminated by some random noise due to experimental error in the measurements of the objective, and statistical estimation techniques must be used to infer the “real” values of the criterion. This is especially encountered in areas such as real-time estimation and control, as well as simulation-based optimization [245]. This is where Monte Carlo simulations come in handy to get approximations of the actual system values. Other statistical estimation techniques include, but are not limited to, the following:

- Stochastic Approximation (SA) [246]
- Finite-difference SA [247]
- Stochastic gradient descent [248]
- Simultaneous perturbation SA [249]

Some stochastic optimization algorithms willingly introduce randomness into their search process in order to accelerate convergence [250] to the global optimum if it exists, and to lessen the sensitivity of the algorithm to modeling errors. Indeed, adding randomness increases the ability of the algorithm to escape a local optimum and eventually to find a global optimum. The randomization principle also makes the algorithm more likely to perform equally well for a wide variety of input data and for many kinds of problems. Stochastic optimization algorithms applying the randomization principle include, but are not limited to:

- Simulated Annealing (SA) to be discussed in a subsequent section
- Quantum Annealing
- Reactive Search Optimization (RSO) [252],[253]
- Random Search [254],[255]
- Stochastic Tunneling [256]
- Stochastic Hill climbing
- Swarm Algorithms such as Particle Swarm Optimization (PSO) to be examined subsequently
- Evolutionary Algorithms to be described in the following section

- Genetic Algorithms
- Evolution Strategies

Stochastic optimization plays an important role in the analysis, design, and operation of systems encountered in a wide variety of disciplines such as statistics, aerospace engineering, traffic engineering, medical science, and business, among a large array of others. Stochastic optimization techniques provide a way to cope with inherent system randomness, non-linearity, and large dimensionality where classical deterministic optimization methods turn out to be inefficient. Stochastic optimization methods typically work by minimizing a scalar-valued loss function, also called performance measure, objective function, measure-of-effectiveness, fitness function, or criterion, over a domain of allowable values for the vector of design variables for the problem. Although stochastic optimization is usually formulated as a minimization algorithm, a maximization problem can simply be transformed into a minimization problem by changing the sign of the objective function. Contrary to deterministic optimization where perfect knowledge about the loss function is available and used to determine the direction of search at every step of the algorithm, stochastic optimization applies to problems where there is some random error in the measurement of the objective function and some random (Monte Carlo) approximation is made to determine the direction of search as the algorithm iterates towards a potential solution [255].

Finally, as any optimization technique, stochastic optimization presents some limitations [255],[257]. The first issue concerns the noisy information about the objective function which induces some resulting error in the output of the stochastic optimization algorithm. The error in the solution decreases as the inverse square root of the number of objective function evaluations, and thus can only be reduced by increasing, sometimes significantly, the number of measurements of the loss function which is fed into the optimization algorithm. In this context, it is generally very cumbersome (sometimes impossible) to systematically indicate when the optimization algorithm has converged close enough to the solution and can therefore be stopped. Indeed, there always exists the possibility that the actual optimal solution lies in some yet unexplored region of the search space. This is especially true when the objective function measurements are noisy, but this may also be true when the objective function involved is simple and well-behaved [258]. Another shortcoming of multivariate stochastic optimization resides in the curse of dimensionality. As the number of dimensions of the problem increases, the volume of the search region increases, and the optimization algorithm becomes less efficient at searching for a global optimum. Furthermore, practical applications often involve a set of constraints on the problem variables that sometimes needs to be handled differently than the allowable search space using ad hoc practical methods specially tuned to the problem at hand [259]. This results in either a wide array of stochastic optimization methods to choose from when trying to optimize a given problem, or the development of again another new practical method to handle the specific problem under study. This leads to the No Free Lunch (NFL) theorem which supports the intuitive statement that there "is a fundamental tradeoff between algorithm efficiency and algorithm robustness (reliability and stability in a broad range of problems)." In other words, algorithms that turn out to be extremely efficient for a certain type of problems may be completely ineffective for other kinds of problems. In the same way there rarely is a "universally best" solution for a general problem, there can never be a "universally efficient" search algorithm [260].

2.5.2. Evolutionary Optimization

The general idea of evolutionary computing was introduced by Rechenberg in the 1960's [261]. His ideas were then carried on by several other researchers who developed various evolutionary algorithms aimed at mimicking the biological processes at the root of evolution. Evolutionary algorithms have the advantages of finding a global optimum without being trapped in local optima, and handling nonlinear and discontinuous problems with large numbers of variables. However, due to their stochastic nature, evolutionary algorithms require a great number of iterations to get significant results and consequently, have their performance measured in terms of speed of convergence. Finally, the problem of premature convergence of the best individuals of the population to a local optimum is a well known drawback frequently found in these techniques [262].

2.5.2.1. Genetic Algorithm (GA)

Genetic Algorithms (GAs) were invented by Professor John Holland and developed by himself and his students at the University of Michigan in the 1960's and 1970's [263]. When inventing GAs, Holland had two ideas in mind: (1) to encode the properties of natural systems into artificial systems in order to (2) improve the understanding of natural adaptation processes. In essence, GAs are adaptive heuristic search algorithms based on the mechanics of natural selection, genetics (crossover and mutation), and evolution inspired from the principles first laid down by Charles Darwin of survival of the fittest. Hence, they can be thought of as an intelligent exploitation of a random search within a defined search space associated with a specific problem.

The basic principle of GAs is to work on the genetic pool of a population of individuals, in order to find the solution, or a potentially better solution, to a given adaptive problem. This method is especially effective since it not only considers the role of mutation, which randomly improves the algorithm, but also utilizes the principle of recombination, or crossover, of chromosomes to improve the capability of the algorithm to approach, and eventually find, an optimum solution [264]. As such, GAs are one of the most effective evolutionary algorithms: they use simulated evolution of a population of individuals in order to "breed" computer programs to find solutions to optimization or search problems. GAs are also considered to be one of the best ways to solve a problem for which little is known: they are very general and usually work for any type of search space.

Since their introduction by Holland, GAs have been widely studied, experimented and applied to various fields of engineering problems. This is essentially due to their ability to provide alternative methods to solving problems and to consistently outperform traditional optimization methods in most of the cases. For most of the real world problems, optimization means finding a set of optimal parameters. Although traditional methods usually perform poorly in such cases, GAs are ideal candidates for the task. GAs evolve in an environment in which a very large set of candidate solutions to the problem lies. The search space may also be composed of several "hills" and "valleys." In most environments, GAs are able to find an optimum solution to the problem. Nevertheless, in some cases, they can be greatly outclassed by more situation specific algorithms. Sometimes, GAs can be computationally expensive, which makes them not always feasible for real time use. Consequently, GAs are not always the best choice for optimizing a problem. They are however one of the most powerful algorithms able to create relatively quickly high quality solutions to a problem [265].

There is no rigorous definition of "genetic algorithm" that actually distinguishes GA from other evolutionary computation methods. Nonetheless, one can easily say that the term "genetic algorithm" encompasses methods that have at least the following properties in common: populations of chromosomes, selection according to a fitness function, crossover to produce new individuals, and random mutation of new offspring [266]. GAs are based on the following important operators [267], [268]:

- Selection: this operator selects individuals in the population for reproduction, according to a fitness function: the fitter the individual, the more likely it is to be selected to reproduce.
- Crossover: this operator randomly selects a starting location and an ending location in the two fittest chromosomes, and exchanges the sub-sequences between these locations to create two offsprings. Crossover is characterized by a probability which gives the likelihood of the selected individuals to be crossed-over. A low probability means that the fittest individuals are most often crossed-over to produce offsprings, while a high probability means that the fittest individuals are directly transferred as offsprings to the next generation without being crossed-over.
- Mutation: this operator is applied to the offsprings of the two selected fittest individuals, and randomly flips some of the components or bits of their chromosomes. Mutation can be applied to every bit of a chromosome or to one or more random bits of each chromosome in the population. Mutation is characterized by a probability which quantifies the likelihood with which a given bit may be flipped. A low probability means that the bit has a low chance of being flipped, while a high probability means that the bit has a high chance of being flipped.

In this context, a genetic algorithm starts with the creation of a population of randomly generated individuals (solutions) represented by chromosomes. The chromosomes should somehow contain the information about the solutions they represent. Therefore, the chromosomes are then encoded, most commonly in a binary string. In this case, each chromosome in the population is a binary string and each bit in the string represents some characteristic of the solution. Of course, there are many other ways of encoding chromosomes, for instance using integers or real numbers. This mainly depends on the problem to be solved. The individuals in the population are then evaluated according to a pre-specified evaluation function, called the fitness function, which gives a score to each individual based on its performance at a particular task. Then, the fittest individuals are selected to "reproduce." These individuals, called parents, are crossed-over to create one or more offspring(s), according to a crossover probability or crossover rate. This process of "reproduction" is motivated by the hope that the new population will be better than the old population, with regards the fitness function. Finally, the offsprings are randomly mutated to create a new population, according to a mutation probability or mutation rate. This process continues until a suitable solution to the problem has been obtained or a certain number of generations have passed [267]. A simple GA is detailed in Appendix B.

There exist several models for implementing reproduction of the population in a GA. The most common are "proportional representations," especially the "roulette wheel," the "rank selection" and the "elitist approach," and the "tournament selection" [269]. Each of these methods is described in Appendix B.

GAs have been and are still currently applied to a large variety of scientific and engineering problems and models. Some examples of application include, but are not limited to [266]:

- **Optimization** (numerical and combinatorial)
- **Automatic programming** (evolution of computer programs for specific tasks, design of other computational structures)
- **Machine learning** (classification, prediction, determination of weights for neural networks)

- **Economics** (modeling of innovation processes, development of bidding strategies, emergence of economic markets)
- **Immune systems** (various aspects of natural immune systems)
- **Ecology** (symbiosis, resource flow)
- **Population genetic** (conditions for evolutionarily viable genes)
- **Evolution and learning** (interaction between individual learning and species evolution)
- **Social systems** (evolution of cooperation and communication in multi-agent systems)

Because of their success in the above and other areas of study, GAs have gained a growing interest among researchers in many disciplines, such as filtering, noise control, computational intelligence, speech recognition, production planning and scheduling, communication systems, distributed sensor networks, and many others [267]. However, there is no rigorous way to know whether GA is a good method to use for a given problem. For instance, if the search space is large, is known to be multimodal and non smooth (i.e. consists of several local optima and of a single smooth global optimum), or is not well defined, or if the fitness function is noisy, and if it only requires to quickly find a sufficiently good solution to the problem, then a GA will have a good chance of being competitive with, or even surpassing, other weak optimization methods that are not domain specific. On the contrary, if the search space is large, is smooth and unimodal, then a gradient-based method, such as the steepest ascent or hill climbing, will be much more efficient than a GA in exploiting the smoothness of the search space. If the search space is well defined, then domain-specific methods will often outperform GAs. If the fitness function is noisy, methods such as simple hill climbing, which search the space according to a one-candidate-solution-at-a-time procedure, might be mistaken by noise, whereas GAs are thought to perform robustly in the presence of small amounts of noise due to the accumulation of fitness statistics over many generations. GAs also work well on mixed (continuous and discrete) combinatorial problems. Most real-world problems involve multiple conflicting objectives where improving one objective may deteriorate the performance in terms of one or more other objectives. In particular, the problem of design and optimization of detection architectures for the surveillance and protection of critical assets and populations as studied in this research is highly multi-modal, is mixed, and involve two conflicting objectives. Therefore, a modified GA has been developed, tested, and applied to the above problem to determine detection solutions. More generally, many heuristic algorithms have been developed to solve multi-objective optimization problems including genetic algorithm and simulated annealing, to be described next. However, it has been proved that GA is an intelligent optimization algorithm able to balance the tradeoff between exploration and exploitation among other significant advantages [270].

The aforementioned statements are only intuitions, and do not rigorously predict the performance of GAs when applied to any specific problems, compared to other search procedures. Indeed, the efficiency of a GA highly depends on the method employed to encode candidate solutions (binary encoding, many-character and real-valued encodings, tree encodings), on the operators, on the parameters settings, and on the particular convergence criterion [266]. That is why, when applying GA to the problem of DODA, it is necessary to perform a rigorous analysis involving:

- The choice of an appropriate reproduction method
- The selection of analytic test functions presenting similar properties to the problem of interest
- The determination of appropriate operators, parameter settings (mutation rate, population size, etc), and convergence criterion using the analytic test functions
- The application of the resulting GA to the initial DODA problem

More details about the advantages and shortcomings of GA may be found in Appendix B.

2.5.2.2. Simulated Annealing (SA)

Simulated Annealing is a probabilistic optimization method adapted from the Metropolis-Hastings algorithm [276], a Monte Carlo method to generate sample states of a thermodynamic system. More details about the construction of the SA algorithm may be found in Appendix C. SA has been proposed independently by Scott Kirkpatrick, Daniel Gelatt, and Mario Vecchi in 1983 [274], and by Vlado Cerny in 1985 [275], in order to find the global minimum of a multimodal cost function (that may thus possess several local optima). In essence, SA is a generic probabilistic method used to optimize a problem by iteratively trying to improve a candidate solution according to a given objective function or measure of quality. The method makes few or no assumptions about the problem being optimized, and is able to search very large spaces of candidate solutions for a good approximation of the global optimum of a given objective function. Nevertheless, it does not guarantee an optimal solution is ever found. SA is often used for discrete search spaces as well and, in certain cases, can be more effective than systematically enumerating all possible candidate solutions and checking whether each candidate satisfies the problem's statement. This is especially true if the goal is to find an acceptably good solution to an optimization problem in a reasonable amount of time, rather than the best possible solution.

SA has been applied to a variety of combinatorial optimization problems, including circuit partitioning and placement [278],[279],[280],[281],[282], strategy scheduling [283],[284] for capital products with complex product structure, umpire schedule in US Open Tennis tournament [285],[286], event-based learning situations [287],[288], image processing [289],[290],[291], Boltzmann Machines [292],[293], graph partitioning [294],[295], graph coloring [296],[297],[298], number partitioning [298],[299], the Traveling Salesman Problem (TSP) [300], and many others [301],[302],[303]. In most problems, it appears that the cooling schedule of the SA algorithm significantly influences the quality of the solution obtained. For instance, in the graph partitioning problem, the quality of the solution can differ as much as 10% depending on the type of cooling schedule employed (exponential, linear, logarithmic, static, dynamic, etc). Finally, despite their relative ease of implementation and their ability to provide reasonably good solutions for many combinatorial optimization problems, SA algorithms often require excessive computational times and a careful choice of their tunable parameters [304].

To conclude, when it comes to multimodal optimization problems, SA algorithms usually perform better than greedy algorithms. Furthermore, SA guarantees convergence towards an acceptable solution after a sufficiently large number of iterations. Nonetheless, due to the dependence of the obtained solution on the cooling schedule, and the excessive computational times sometimes observed, various alternatives to the traditional SA algorithm have been proposed. For instance, in 1986, Bohachevsky, Johnson, and Stein [305] proposed a generalized SA procedure for continuous optimization problems, and applied it to an optimal design problem. Other variants of SA based on Bayesian ideas have been suggested in 1989 by Laud, Berliner, and Goel [306], and by Van Laarhoven et al. [307]. Finally, SA is related to a variety of other methods, including, but not limited to:

- ✓ Stochastic tunneling which overcomes the difficulty of SA in escaping from local minima as the temperature decreases, by "tunneling" through energy barriers.
- ✓ Stochastic gradient descent which runs many greedy searches from random initial states.
- ✓ Tabu search which preferentially transitions to lower energy states, but can occasionally transition to higher energy states when it is stuck in a local minimum. It also avoids going back to a previously "visited" state by keeping a *"taboo list"* of already seen solutions.
- ✓ Genetic algorithms which work on a population of solutions rather than just a single one.
- ✓ Graduated optimization which smoothes the target function while optimizing it.
- ✓ Ant colony optimization which uses many agents to explore the solution space and find locally productive areas.
- ✓ Particle swarm optimization which models the intelligence of a swarm to find a solution to an optimization problem, or to model and predict social behavior in the presence of objectives.

2.5.2.3. Tabu Search

Tabu search is a metaheuristic local optimization algorithm created by Glover in 1986 [308] and formalized in 1989 [309],[310]. It is based on a neighborhood search procedure described in details in Appendix D. It starts from a potential solution to the problem and iteratively examines its neighbors in order to find an improved solution, until some stopping criterion is satisfied or some fitness threshold is reached. Conceivably, local optimization techniques often have a tendency to get stuck in suboptimal regions or on plateaus where several solutions can be equally fit. In order to avoid such pitfalls and explore the search space more thoroughly, tabu search introduces memory structures that describe all the solutions visited at any given step of the optimization process or that contain user-prescribed sets of rules [309]. If, during the search, a potential solution has been previously encountered by the algorithm, or if it has violated a rule provided by the user, it is marked as "taboo" and cannot be re-admitted to the neighborhood of the current potential solution as a potential improved solution. The memory structures therefore construct a tabu list of solutions that have been visited in the past. They can be short-term, intermediate-term or long-term structures as follows [311]:

- Short-term memory structures contain a list of solutions that have been recently visited. If a potential solution belongs to this list, it cannot be re-visited until the short-term memory expires.
- Intermediate-term memory structures are composed of a list of rules that bias the search towards areas where promising solutions tend to be located. For instance, the rules may prohibit certain types of solutions or certain moves that would lead to undesirable solutions. Such solutions have so-called "tabu-active" attributes that must be banned from the search over a given period of time.
- Long-term memory structures are formed of rules that intensify and diversify the search process by resetting the solution when the algorithm gets stuck in a suboptimal region or on a plateau.

Although short-term memory structures are generally enough to reach a solution presenting better fitness properties than those found by conventional local search algorithms, intermediate-term and long-term structures may be necessary when considering complex optimization problems.

Like any other local optimization technique, tabu search has some major issues. The first one is that it is only effective in discrete space, and it has some difficulty dealing with large or highly-dimensional problems where it tends to explore only a small portion of the search space. In order to extend the exploration capability of the algorithm, some implementations of the tabu search focus on some specific attributes of the solution, rather than its entire form, and create a tabu list of these attributes that need to improve. Nevertheless, this type of memory structures that contains a list of one or more attributes to avoid rather than complete solutions, tend to eliminate, at least temporarily, solutions that might have excellent overall qualities, although they contain the tabu-active attributes. In order to mitigate this effect, some tabu search algorithms use "aspiration criteria" to override the tabu state of a solution that has a better fitness than the current best solution, but that would otherwise be excluded from allowable set. Such a solution is thus allowed to be visited temporarily even though it contains banned attributes [312].

To conclude, tabu search has successfully been applied to various fields of study including, but not limited to the following:

- Company management (scheduling, assignment) [308],[314],[315],[316]
- Telecommunications [317]
- Probabilistic problems [318]
- Neural networks [319]
- Traveling salesman problem [320]
- Graph theory [321]
- Flow shop problems [322],[323]
- Electronics [324]
- Non-convex optimization [325]

2.5.2.4. Ant Colony Optimization (ACO)

Ant Colony Optimization is a metaheuristic probabilistic technique developed by Marco Dorigo in 1992 [326],[327] to solve computational problems aiming at finding optimal paths in graphs. It belongs to the family of swarm optimization methods and is based on the behavior of ants exploring paths between their nest and sources of food. The basic idea lies in the abilities of ants to collectively find the shortest path between a source of food and the colony using their individual limited cognitive skills. The underlying optimization process is described in Appendix E.

In the realm of computational problem optimization, the ants searching for food by laying down pheromones along the paths they explore between the nest and a food source model the individual optimization agents searching for a global optimum by communicating with each other as they explore the graph representing the problem to solve [330]. In this context, pheromone evaporation is a way for the optimization algorithm to avoid convergence to a local optimum. If the pheromones did not evaporate, then the trails followed by the first ants would become increasingly attractive to the nearby ants as more and more ants would be attracted by and would enhance the original pheromone trails. In this case, the exploration of the solution space would be constrained to the regions explored by the first ants. Indeed, the other ants would not be inclined to follow different paths with no pheromone signature, and thus would not search for other more optimal routes to the food source.

Since its creation in 1992, the ACO algorithm has diversified and is now applied to a wide class of problems in computer sciences and operations research. Variations of the ACO algorithm and examples of applications are provided in Appendix E. ACO algorithms are particularly interesting in network routing problems [352] and transportation system because of their ability to continuously adapt to dynamic problem changes in real time. This is an advantage compared to similar evolutionary approaches such as Simulated Annealing or Genetic Algorithm. Furthermore, contrary to other swarm optimization algorithms, ACO algorithms iteratively construct solutions to combinatorial problems. Therefore, it is always possible to find a best solution, even though no ant actually travels the shortest route. The latter can be built from the paths which have the strongest pheromone signature. This can be problematic for real variable problems where the notion of neighbors does not exist. Current research in ACO algorithms is mainly devoted to theoretical foundations and to applications to emerging challenging problems [353],[354],[355],[356],[357]. In 2000, Gutjahr [358] initiated the development of theoretical foundations and was the first to prove the probabilistic convergence of an ACO algorithm. A general description of theoretical results for the ACO algorithm can be found in Dorigo and Blum [359].

Finally, ant colony optimization is one example of a large variety of swarm intelligence algorithms. Another similar algorithm is the Glowworm Swarm Optimization (GSO) algorithm, presented by Krishnanand and Ghose in 2009 [360]. The GSO algorithm shares some features with ACO and PSO to be discussed below, and allows the simultaneous computation of multiple optima of multimodal functions. It involves glowworms emitting light from the luminescent substance they carry, the luciferin, in order to interact with other glowworms and give them information about their current location. Each glowworm is attracted towards the brighter glow of other glowworms in their neighborhood and moves towards one of the neighbors which emits the higher amount of light. The motion of glowworms which depends on the intensity of luciferin owned by their neighbors allows the swarm to partition into disjoint subgroups. This type of algorithm has been used in a variety of optimization problems where the initially obtained solution needs some refinement [144].

2.5.2.5. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) was developed by Eberhart and Kennedy in 1995 [361]. It is a stochastic, unconstrained optimization technique dealing with non-linear functions that has roots in two main component methodologies: Artificial Life (A-life) and swarming theory. PSO is also related to evolutionary computation, and has similarities with both genetic algorithms and evolution strategies. Nonetheless, PSO is different from traditional evolutionary algorithms in that, instead of exploiting the competitive aspects of evolution, it exploits its cooperative and social aspects. PSO was indeed inspired by swarm intelligence behavior, such as bird flocking and fish schooling [362]. It is based on a suitable model of social interaction between independent agents (particles), and it uses social knowledge (or swarm intelligence) to find the global optimum of a generic function. PSO is based on the principle that each individual can benefit from the discoveries and previous experience of all other companions during the search for food. In the PSO algorithm, each companion, or particle, in the population called a swarm, is assumed to "fly" over the search space to find promising regions of the landscape. Unlike any other evolutionary algorithm, PSO does not use evolutionary operators to manipulate individuals of the swarm. Rather, each individual in the swarm flies in the search space with a "velocity" which is dynamically adjusted according not only to its own flying experience, but also to that of its companions [363]. Hence, PSO has the ability to search effectively large spaces.

The PSO algorithm is somehow similar to a genetic algorithm [364] in that it is initialized with a population of random potential solutions called particles. Unlike a GA, however, each potential solution or particle is assigned a random position and a random velocity. Each particle is also characterized by a scalar fitness value derived from the objective function that needs to be optimized. The particles are then flown through the search space such that the "flying" pattern of a given particle is influenced not only by the location of the best solution achieved by the particle itself during its trip, but also by the location of the best solution achieved by the population as a whole. These best solutions are called particle best and global best respectively [363], [365], [366]. Furthermore, in PSO, the addition of a velocity to the current position of a particle to generate its next position resembles the mutation operation in GA, except that "mutation" in PSO is guided by both the particle's flying experience and the group's flying experience. As such, PSO can be said to perform *"mutation with a conscience."* Besides, by looking at the particle's own best position obtained so far as additional population members, PSO also presents a form of selection even though it is relatively weak [367]. Finally, because there is no crossover operator in PSO, each individual in an original population has a corresponding partner in a new population. As such, PSO can to some extend avoid the premature convergence and stagnation observed with GAs [368]. In short, the basic PSO algorithm may be stated as follows:

1. Define the optimization problem to be solved, i.e. the objective function to optimize, and determine the search space.
2. Create a population (swarm) of random particles uniformly distributed over the search space and with initial random velocities.
3. Evaluate the position of each particle of the swarm according to the objective function.
4. If a particle's current position is better than its previous best position, then update it.
5. Determine the best particle of the swarm according to the particles' previous best positions (the swarm's best position can be defined as the best of the particles' best positions).
6. Update the particles' velocities according the particles' bests and the global best.
7. Move the particles to their new positions.
8. Go back to step 3 until the stopping criteria are satisfied (good enough solution or maximum number of iterations).

Some additional comments concerning the local and the global characteristics of the PSO algorithm may be found in Appendix F.

One of the advantages of PSO is its simplicity. Indeed, at each iteration, the particles' positions and velocities are updated according to:

- The particles' positions at the previous iteration
- The particles' velocities at the previous iteration
- Learning factors or acceleration parameters (cognitive and social parameters)
- An inertia weight parameter
- The particles' bests at the previous iteration
- The global best at the previous iteration

The inertia weight is employed to control the impact of the previous history of the particle on its current behavior. Accordingly, this parameter regulates the trade-off between the global (wide-ranging) and the local (nearby) exploration capability of the swarm. A large inertia weight facilitates global exploration, while a small one tends to facilitate local exploration. Consequently, it is worth making a compromise for the value of the inertia weight such that the exploration capability decreases as the swarm is evolving. The swarm therefore has a high global exploration capability during the first iterations so as to find the region in which the global optimum is located, and then, when this region has been determined, the swarm fine tunes the search area around the global optimum so as to locate it more precisely.

To summarize, PSO has good performance, low computational cost and is easy to implement. Due to its population-based solutions mechanisms, PSO is suitable for multi-disciplinary optimization. It is indeed capable of providing several solutions in one execution, in contrast to traditional techniques where one execution is capable of providing one single solution. The reason for the high quality of the PSO algorithm in certain experiments may be that:

1. The <u>coverage of the search space</u> is random and wide. Therefore, an increase in population size has a greater probability of reaching the global optimum at an early stage of the search.
2. The <u>global nature of the search</u> offers insight into various local neighborhoods of the search space.
3. Particles moving fast towards the best particle of the swarm allow PSO to perform <u>detailed search</u> of a good region at an early stage.

However, these properties of PSO can also result in locating a local optimum. To alleviate this problem, Parsopoulos et al. suggested a modified PSO, which has been shown to exhibit improved performance in numerous experiments related to classic optimization problems [380], [381], [382], [383].

Last but not least, PSO are similar to GAs, and therefore are able to solve many of the same kinds of problems. However, PSO does not suffer from the same difficulties as GAs. In PSO, progress towards the solution is enhanced and not detracted by interaction between particles (individuals) of the group (population). Moreover, a particle swarm system has a memory: each particle keeps both the memory of its own best position and of the group's best position. In PSO, individuals who fly past optima are made to return towards them since knowledge of good solutions is retained by all particles. On the contrary, changes in the genetic populations in GAs result in the destruction of previous knowledge of the problem, except when the best individual or the first several best individuals of a given generation is/are automatically passed to the next generation through the elitist approach. In the later case, one or a small number of individuals effectively retain their "identities" (or memories) as they are passed to the next population [361]. This may explain why PSO has successfully been applied to many areas of research, including artificial neural network for evolving connection weights during training [368], [384], design of controllers [385], optimization of biochemistry processes [386], optimization of power flow [387], tuning of controller parameters [388], dynamic bifurcation analysis of chemical processes [389], optimization of the Traveling Salesman Problem [390], hydraulics [391], networked sensor systems [392], image classification [393], design of combinatorial logic circuits [394], automated operations [395], flowshop scheduling [396], and many others.

2.5.2.6. Pareto Optimization

Pareto optimality is a concept named after the Italian engineer, economist, sociologist, political scientist, and philosopher Vilfredo Pareto [397],[398], initially developed for studies of economic efficiency and income distribution. In its economic application, Pareto efficiency refers to an economic allocation in which no one can be made better off without degrading at least one individual. A Pareto improvement corresponds to a change in the allocation of goods among a group of individuals which makes at least one individual better off without making any other individual worse off. When no further Pareto improvement is possible in an economic allocation, then the allocation is said to be

“Pareto efficient” or “Pareto optimal”. In the context of economics, Pareto efficiency does not necessarily mean that the distribution of resources is socially desirable, equitable, or even adapted to the well-being of the society [399],[400].

Pareto optimality has also been largely applied to engineering problems where the goal is to select efficient alternatives among a set of design solutions. Each potential option is assessed under a set of criteria and a portfolio of Pareto optimal options whose members are not categorically outperformed by any other option members is identified. Given a group of choices and a way of assessing them, the Pareto frontier or Pareto front is the ensemble of choices that are Pareto efficient. Instead of evaluating the full range of options, a designer can then restrict the study to the group of choices that are Pareto efficient. By definition, the Pareto frontier is the set of feasible designs that are not strictly dominated by any other point. The general definition of Pareto optimality is provided in Appendix G.

Engineering problems most often involve multi-criteria optimization in which the notion of Pareto optimality comes in handy. Solving multi-objective optimization problems with conflicting objectives usually leads to a set of non-dominated solutions rather than a unique solution. This set of non-dominated solutions, which can be continuous, discontinuous, smooth, or non-smooth, is the Pareto front. In general, when the problem involves highly non-linear objectives and constraints, or multiple disciplines, the Pareto frontier cannot be determined analytically, and obtaining a single Pareto solution can be very time consuming due to the complexity of the problem [403].

For many years, engineering optimization has been characterized by a single objective function, although this is not a rigorous assumption for most real-world problems which usually involve multiple, potentially conflicting, objectives. In this context, most of the design tradeoffs made at that time were mainly based on experience, rather than on some optimal criterion [404]. However, multi-objective optimization problems typically present a solution space composed of alternatives which are superior to the rest of the solutions but which may be inferior to other solutions in one or more objectives when every criterion is considered. Such Pareto optimal solutions can be defined as being better than any other solution but cannot be distinguished between each others. Therefore, the goal of multi-objective optimization is to be able to obtain as many Pareto optimal solutions as possible. Once the complete set of Pareto efficient solutions has been found, higher-level decision-making considerations must be considered to qualify one of them as the optimal solution for the problem under study. When trying to solve multi-criteria optimization problems, classical methods such as weighted sum, goal programming, and min-max are not efficient because of their inability to find multiple Pareto optimal solutions in a single run. Indeed, they need to be applied as many times as the number of desired Pareto optimal solutions and multiple evaluations of these methods do not guarantee the quality of the obtained Pareto solutions. On the contrary, because they use populations of solutions in their search, evolutionary optimization algorithms have proven to be able to find the full set of Pareto optimal solutions in a single run. Several algorithms have been developed in that perspective and are now widely used to determine the set of Pareto optimal solutions for multi-criteria optimization problems [415],[416],[417],[403],[405],[406],[407],[408],[409]. Some of them are described in Appendix G.

CHAPTER III

RESEARCH QUESTIONS, HYPOTHESES, AND PROPOSED APPROACH

3.1. Summary of the Problem and Proposed Methodology

As thoroughly mentioned in previous sections, the surveillance of geographic borders, critical infrastructures, or large areas using limited sensor capability has always been a challenging task in many homeland security and distributed sensor network applications. Indeed, geographic borders may be very long and may go through isolated areas that are sometimes uninhabitated. As for critical assets, they may be large and numerous and may be located in highly populated areas. As a result, it is virtually impossible to secure both each and every mile of border around the country, and each and every critical infrastructure inside the country. Most often, a compromise must be made between the percentage of border or critical asset covered by surveillance systems and the induced cost.

Although threats to homeland security can be conceived to take place in many forms, those regarding illegal penetration of the air, land, and maritime domains under the cover of day-to-day activities have been identified to be of particular interest by the American government and by several European governments. For instance, the proliferation of low altitude aerial systems, combined with regular air traffic growth, poses a unique challenge for the surveillance of homeland airspace and in particular for identifying potentially hostile vehicles interoperating with friendly aircraft. Similarly, the proliferation of drug smuggling, illegal immigration, international organized crime, and more recently, modern piracy, require the strengthening of land and maritime borders awareness. Hence, land border and maritime intelligence assessments point to increasingly complex and challenging national and coastal security environments. In this context, the ability to monitor, detect, identify, and eventually intercept suspicious entities or systems well before they reach the border or strategic land and coastal sites is of critical importance to prevent dangerous activities from jeopardizing populations and governments. Nevertheless, suspicious aerial vehicles, persons, ground systems or maritime entities can dexterously hide amongst and efficiently interoperate with friendly persons and systems so as to compromise the situational awareness of border and maritime protection missions. This calls for security improvements in the aerial, border and maritime domains while preserving prosperity and minimizing disruptions or delays to commerce and global trade. As a consequence, it is necessary to comprehensively understand what composes each of the aforementioned domains, which entails acquiring accurate knowledge about the movement of aircraft, ground vehicles, marine vessels, cargo, and people.

The complexity and challenges associated to the above mission and to the protection of the homeland may explain why a methodology enabling the design, modeling, simulation, and optimization of detection architectures or networks of distributed sensor systems, able to provide accurate scanning of the air, land, and maritime domains, in a specific geographic and climatic environment, is a capital concern for the defense and protection community. As a result, the present work focuses on the development of adequate architectures of ground platforms and sensors for monitoring aerial, land, and maritime systems. This primarily involves identifying the best combination and positioning of detection and surveillance systems able to monitor the homeland and its shores, and to collect information about the surrounding aerial, terrestrial and maritime environments. To do so, it is imperative to quantitatively assess current state-of-the-art as well as future notional systems, generate meaningful comparisons across disparate platforms, explore tradeoffs between a myriad of factors, and identify key technological gaps. However, this effort is riddled with difficulties inherent in the definition, analysis, and assessment of systems-of-systems (SoS) where a variety of distributed heterogeneous systems actively interact to generate a top-level capability [426]. In fact, it should be recognized that most DODA applications share a majority of these challenges [11], [427], [428]. One of them is concerned with the large number of possible platforms that can be considered to compose the detection architecture. Many of these may offer overlapping functionalities or may not be directly comparable altogether. Another issue deals with the importance of an operational context in terms of how it drives the physical and functional architecture of the detection SoS, and the performance requirements of each of its platforms. For instance, the physical placement of detection systems within the architecture and their need to communicate among themselves is driven by the terrain, the weather, and other environmental conditions. In turn, said environmental factors may translate to performance requirements for individual platforms in the architecture or to constraints in the global structure. These and other related challenges vastly expand the combinatorial nature and the complexity of the problem. This represents in itself a hindrance to the analyst who seeks to formulate and understand the problem construct, conduct an exhaustive study of all available possibilities, and present key tradeoffs in a transparent fashion. Often times, this leads to an unintentional overreliance on expert opinion and past experience. While recognizing their importance, the objective of the defense community should be to formulate flexible processes that leverage on established analytical techniques and practices while incorporating necessary modifications where required.

This thesis proposes a methodology aimed at addressing the aforementioned gaps and challenges. The Modeling, Simulation and Optimization of Distributed Detection system Architectures or M-SODDA methodology particularly reformulates the problem in clear terms so as to facilitate the subsequent modeling and simulation of operational scenarios of interest. The needs and challenges involved in the proposed study are investigated and a detailed description of a multidisciplinary strategy for the design and optimization of distributed detection system architectures in terms of detection performance and cost is provided. First, a multi-level morphological decomposition of the problem is performed to identify its main elements, and investigate relevant sub-elements, alternatives and attributes able to capture the various facets of the homeland security mission of interest. Second, the pair-wise compatibilities between the various elements, sub-elements, alternatives, and attributes identified, are assessed so as to obtain consistent and realistic, yet notional, operational scenarios. Third, a framework for the modeling and simulation of previously selected scenarios is created, and improved methods for the rapid optimization of detection architectures in specific operational situations are developed. More precisely, the present thesis describes a new approach to determining detection architectures able to provide effective coverage of a given geographical environment at a minimum cost, by optimizing the appropriate number, types, and locations of surveillance and detection systems on the theater of operations. In this study, the physical design parameters of the sensor systems are either specified or allowed to vary within predefined ranges, and the customer preference with respect to performance and cost is captured by a parametric benefit to cost analysis encapsulated in the objective function to optimize. The goal of the optimization is twofold. First, given the topology of the terrain under study, several promising locations are determined for each type of detection system based on the percentage of terrain it is covering. Second, architectures of distributed, fixed sensor systems able to effectively cover large percentages of the terrain at minimal costs are determined by optimizing the number, the types and the locations of each detection system in the architecture. To do so, two evolutionary optimization approaches (Genetic Algorithm (GA) and Particle Swarm Optimization (PSO)) are selected to solve the homeland security mission of interest. Their structure is modified compared to their original versions, and their main parameters are tuned to satisfy the specific requirements of the optimization problem. The convergence properties of the modified optimization approaches are then compared, and the algorithm presenting the best performance at the lowest computational cost is selected for further study. Finally, a recursive heuristic optimization algorithm is developed to obtain benchmark detection architectures and to assess the ability of the previously selected modified optimization algorithm to provide consistent results and accurate solutions to the original homeland security application.

Because of the various sources of uncertainty involved in engineering design, an optimized solution may turn out to be suboptimal if not infeasible. In this work, sensitivity analysis is used to study the impact of perturbations in the design variables on the optimized detection architectures. Indeed, the optimization process yields a portfolio of fixed detection architectures, composed of detection systems that are located at specific positions on the theater of operations. However, the performance and the cost of the optimized detection architectures are direct functions of their structure and their composition. Indeed, such criteria highly depend on the number, the types, the properties, and the positions of the detection systems constituting the detection architectures. In order to determine whether these optimized detection architectures are able to perform under real-world situations, they first need to be incorporated into the modeling and simulation framework and to be exercised in a simulation where items of interest are actively detected by the architectures. This enables the user to identify potential gaps of detection in the simulated architectures and to bridge these gaps with mobile detection systems that are transported by patrol units. Once the performance of the fixed detection architectures has been enhanced by mobile detection systems when and where necessary, the robustness of the resulting architectures may be evaluated. This may be done by exercising the resulting detection architectures under various operational scenarios in the modeling and simulation environment. This may also be done through a sensitivity analysis in which the number, the types, the properties, and/or the positions of fixed and mobile detection systems in the detection architectures are changed, and the impacts of these changes on the performance and cost are investigated. This sensitivity analysis provides insights into the influence of the structures and the compositions of the detection architectures on their performance and cost, and enables the user to obtain a portfolio of detection architectures, composed of both fixed and mobile sensor systems, able to properly detect items of interest in the specific operational environment considered at a minimal cost.

To conclude, the M-SODDA methodology provides a structured, traceable, and reproducible way of obtaining coverage- and cost-efficient detection architectures able to properly monitor large areas in specific operational environments. It is not meant to yield a fully implementable and deployable surveillance and detection system-of-systems that has been validated and verified in real-world situations. Therefore, it does not target the operational agents actively operating and maintaining the detection systems on the terrain. Rather, the proposed methodology is structured around a set of methods and tools that make it the perfect fit for government industries as they try to demonstrate their capabilities to potential decision makers in the field of homeland security, and for government entities as they solicit the industry to perform a study of their needs and to suggest solutions to their requests. Indeed, the proposed methodology provides the industry managers and decision makers with a means to market their products or services and to respond to the specific needs of a potential governmental user by demonstrating their ability to design and optimize detection architectures featuring one or more of their own systems. It also provides the governmental manager or decision maker with an integrated framework for understanding the process of designing and optimizing detection architectures for surveillance and protection missions of interest. With this framework, the decision maker may play any kind of "what-if" scenarios on the suggested architectures in order to generate one or more detection architecture solutions satisfying specific expectations and operational constraints.

3.2. Research Objectives, Research Questions and Hypotheses

3.2.1. Research Objectives

The information contained in the previous chapters help identify the main objectives of this research. These are the following:

- Homeland security surveillance and protection missions
- Detection of various types of items of interest
- Discovery of a portfolio of distributed sensor architectures featuring both fixed and mobile detection systems
- Optimization of the number, types, properties, and positions of detection systems on the terrain, given geographic and climatic conditions
- Performance and cost of detection architectures
- Formulation of a structured, traceable, and reproducible methodology that features flexible processes leveraging on established analytical techniques and practices while incorporating necessary modifications where required

3.2.2. Observations, Research Questions (RQ) and Hypotheses (H)

In this section, several observations are made, accompanied by research questions and hypotheses addressing the problem of designing, modeling, simulating, and optimizing distributed detection system architectures for specific homeland security missions.

<u>OBSERVATION</u>: New threats to the homeland are continuously emerging and may be hard to detect in our modern, busy societies. In order to be able to adapt its response to the growing number of potentially harmful people and systems, the defense and protection community needs to be able to timely, properly and effectively detect what could be a threat to the homeland. As a result, the development of efficient detection architectures for airborne, ground, and/or maritime threats to critical assets and populations is a major priority.

<u>OBJECTIVES</u>: To do so, it is imperative to quantitatively assess current state-of-the-art as well as future notional systems, generate meaningful comparisons across disparate platforms, explore tradeoffs between a myriad of factors, and identify key technological gaps.

<u>OBSERVATION</u>: However, a detection architecture is a system-of-systems composed of disparate detection system concepts. Therefore, the effort is riddled with difficulties inherent in the definition, analysis, and assessment of systems-of-systems, where a variety of distributed heterogeneous systems actively interact to generate a top-level capability.

<u>OBSERVATION/CHALENGES</u>: A large number of possible detection systems can be considered to compose the detection architecture. Many of these may offer overlapping functionalities or may not be directly comparable altogether. Also, the operational context is significant since it drives the physical and functional architecture of the detection system-of-system, and the performance requirements of each of its elements. For instance, the physical placement of detection systems within the

architecture and their need to communicate among themselves and with external entities is mainly driven by the terrain features and the weather conditions. In turn, said environmental factors may translate into performance requirements for individual detection systems in the architecture or constraints in the global structure.

These and other related challenges vastly expand the combinatorial nature and the complexity of the problem. This represents in itself a hindrance to the analyst who seeks to formulate and understand the problem construct, conduct an exhaustive study of all available possibilities, and present key tradeoffs in a transparent fashion. Often times, this leads to an unintentional overreliance on expert opinion and past experience.

OVERARCHING RESEARCH QUESTION: How may heterogeneous systems best be distributed over large areas to provide adequate global coverage at a reasonable cost in the context of homeland security? Several examples of homeland security applications of interest are depicted in Figure 10.

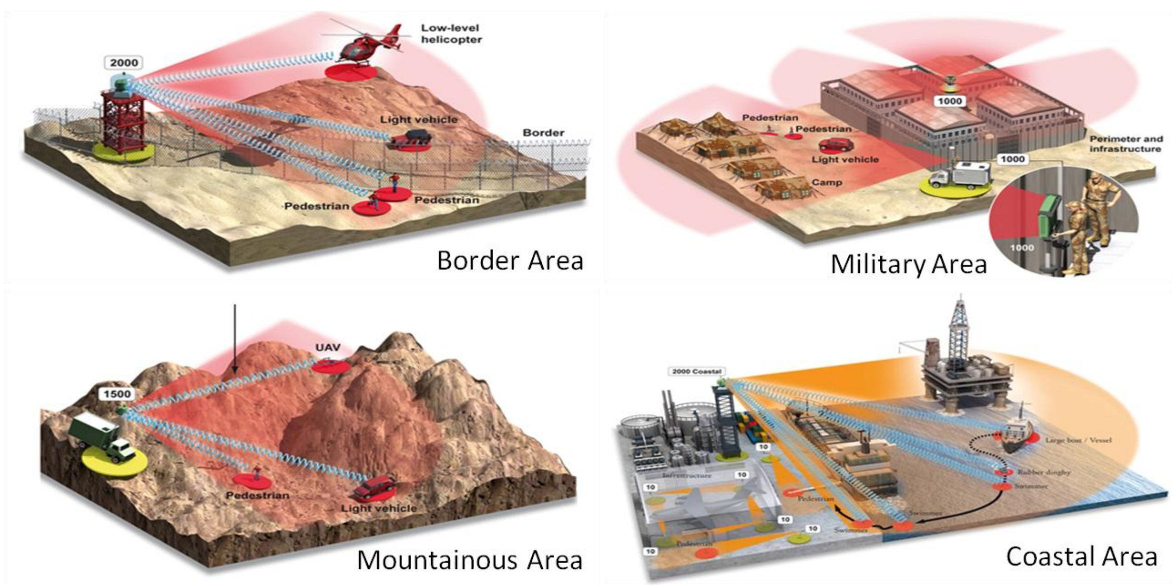


Figure 10: Examples of Homeland Security Applications Requiring the Definition of Distributed Detection System Architectures [429]

OBJECTIVE: Formulation of a methodology that features flexible processes leveraging on established analytical techniques and practices while incorporating necessary modifications where required.

OBSERVATION: In order to formulate a structured methodology for the design and optimization of adequate detection architectures for the protection of critical assets and populations in specific operational contexts, the decision maker first needs to understand the nature of the detection architecture, its composition, and its interactions with external entities and operational conditions.

METHODOLOGY CONSTRUCTION – RESEARCH QUESTION: How can the decision maker truly understand the nature of the distributed system architecture, assess its capabilities, and capture the key tradeoffs between the various elements of the problem?

HYPOTHESIS 1: Physics-based representations of the various elements of the homeland security mission in a modeling and simulation environment will provide the decision maker with a means to play any number of notional "what-if" scenarios through which the capabilities of the systems architecture, as well as the relative sensitivity of tradeoffs at each level of the problem may be assessed.

OBSERVATION: The analyst is ultimately attempting to represent a wide variety of existing detection systems and various entities (people and aerial, ground, or maritime vehicles) identified as being of particular interest. At the same time, the analyst seeks to have the flexibility to generate notional detection systems representative of future advanced concepts. Similarly, it is highly desirable to have the freedom to consider notional threats that had not originally been conceived. This is also applicable in the representation of a wide range of critical assets, located in various operational environments (i.e. various combinations of geographic and climatic conditions).

PARAMETRIC REPRESENTATION – RESEARCH QUESTION: How can the decision maker represent various states of the world through a wide range of both existing and notional entities composing the homeland security mission of interest?

HYPOTHESIS 2: A structured, yet flexible, characterization scheme, built on the concept of *"parametric representations,"* will provide a way to fulfill both the need to characterize a wide range of existing elements, and the incentive to generate new notional ones in a single step.

PROBLEM DECOMPOSITION – RESEARCH QUESTION: How can the generation of parametric representations be facilitated?

HYPOTHESIS 2.1: A structured approach whereby the homeland security problem of interest is progressively decomposed into its main elements, both physically and functionally, will enable the generation of parametric representations by adequately regrouping different elements of the problem, thus revealing sets of common parameters.

MORPHOLOGICAL ANALYSIS – RESEARCH QUESTIONS: How can the functional and physical decomposition of the homeland security problem be structured for parametric representations and subsequent analysis?

Once the elements of the homeland security mission have been parametrically represented, how can the problem be subsequently recomposed, or synthesized, so that top-level capabilities, resulting from lower level representations and interactions, be revealed?

OBSERVATION: The problem spaces for critical assets, detection systems, items of interest, and geographic and climatic environments, need to be analyzed in an explorative and exhaustive fashion.

HYPOTHESIS 2.2: Morphological Analysis (MA) combined with Hierarchical Decomposition methods will provide a robust, rigorous, structured and traceable process to decompose the problem for parametric representations. Both methods combined with fundamental systems engineering concepts of decomposition and synthesis, such as the systems engineering “Vee” diagram, will provide a means to recompose or synthesize the problem so that lower level representations and interactions may be revealed.

OBSERVATION: Though MA is not new, it has certainly benefitted from recent advances in computation. This has led to a number of successful methodological variants across a wealth of contemporary applications such as air transportation systems risk assessment, sabotage and attacks to nuclear power infrastructure, and other major events. Nevertheless, the inherent hierarchical structure of most MA formulations only provides two levels, that is, a single decomposition/synthesis step between a system and its elements. The problem of protection of critical assets and populations, as studied in this research, is composed of several functional and physical levels that need to be captured according to a balance between exhaustiveness and relevance to the current application.

MORPHOLOGICAL ANALYSIS REVISITED – RESEARCH QUESTION: How can the classic MA approach be modified to represent all levels of functional and physical decomposition for the problem under consideration, and to provide a set of alternatives that is neither unmanageable nor incomplete?

HYPOTHESIS 2.3: Incorporating a multi-level approach to the original MA method, will enable the determination of a set of alternatives that best matches all levels of decomposition, and will allow accommodating any successive decomposition steps that may be required, thus more closely following the conceptual formulation of the systems engineering “Vee.”

OBSERVATION: The general MA approach uses a morphological matrix or matrix of alternatives documenting how a system of interest is decomposed into main element classes. It further enumerates, for each element class, all possible alternatives in an iterative process. However, these alternatives may be of various natures and may have different properties. They may also have different scales associated with them (in terms of size, time, etc) and may not be compatible when considered concurrently in a given scenario. In addition, a typical morphological matrix usually contains far too many combinations of alternatives to be inspected by hand.

CROSS-CONSISTENCY ASSESSMENTS – RESEARCH QUESTION: Given the proposed decomposition of the homeland security mission, the parametric representations of the main elements identified, as well as the nature of these elements, how can internal relationships be examined so as to "reduce" the number of scenarios that can be played to an operationally relevant set?

HYPOTHESIS 2.4: Cross-consistency assessment methods will enable documenting relational data between alternatives identified in the problem decomposition, thus establishing the combinatorial logic that drives the problem synthesis into a number of internally consistent operational configurations.

OBSERVATION: Traditional MA uses a binary scale in its cross-consistency assessment to determine whether two elements are compatible (1) or incompatible (0). Each alternative in the Morphological Matrix is compared, in a pair-wise manner, to all of the others, much like a cross-impact matrix. For most problems, with each pair-wise relation, a judgment is made as to whether – or to what extent – the pair can co-exist, i.e. whether it is a compatible or an incompatible relationship. Nevertheless, for the problem under consideration, a simple assessment of compatibilities is not enough and is actually not appropriate. Indeed, it turns out that, most of the time, the question "Is this alternative or concept compatible with this other alternative?" does not have a simple "yes" or "no" answer. Rather, the answer is most likely "it depends" or "well, not really but the alternatives are not strictly incompatible either."

CROSS-CONSISTENCY ASSESSMENTS REVISITED – RESEARCH QUESTION: How can the ambiguity resulting from cross-consistency assessments of alternatives be resolved for the homeland security mission under study, given that the traditional binary scale used to study the compatibilities between alternatives in the original MA formulation is neither sufficient nor appropriate?

HYPOTHESIS 2.5: Cross-consistency assessments based on probabilistic or likelihood representations will provide a way to describe the relative consistencies at each level of decomposition identified in the Morphological Matrix, as well as the coexistence of alternatives in an operational scenario depending on their characteristics. Such cross-consistency assessment schemes will enable encoding relational data with higher resolution scales to capture more complex interactions.

TRANSITION/HYPOTHESIS: With the aforementioned improvements, the proposed MA approach may be used as a mechanism to structure and document the top-down physical and functional decomposition of the detection SoS, as well as the relational data that drives its bottom-up synthesis back to the highest level of the hierarchical structure. At that point, the information gathered along the multi-level morphological decomposition of the problem and the multi-level cross-consistency assessments of its elements, paves the way for the modeling and simulation of candidate scenarios. In this context, a modeling and simulation framework needs to be developed so that all relevant analyses may be conducted while appropriately capturing the relevant constituents of the operational theater. In particular, surveillance of critical assets, populations, and of aerial, ground or maritime borders is a highly complex problem, characterized by constant changes in the environment (items of interest, detection systems, critical assets, and geographic and climatic conditions). Furthermore, the sensitivity of the detection architecture to such changes can have dramatic consequences on the ability to perform efficient surveillance and protection missions as part of homeland security efforts.

MODELING, SIMULATION, AND OPTIMIZATION ENVIRONMENT – RESEARCH QUESTION: How can the decision maker accurately, rapidly and efficiently capture the impact of changes in the operational situation on the structure (composition and design) of the distributed system architecture?

<u>OBSERVATION</u>: Proper detection of items of interest in a specific operational environment, involves considering systems-of-systems and complex non linear dynamic autonomous systems interacting together to perform a capability and reacting/adapting their behavior to changes in their environment.

<u>HYPOTHESIS 3</u>: Physics-based modeling, combined with agent-based modeling will provide a means to develop a modeling and simulation environment enabling the identification of key factors driving the structure of the distributed system architecture according to changes in operational conditions. Such modeling capabilities, complemented with the modified MA method, will allow the definition of a structured, robust, rigorous and traceable process for the simulation of notional homeland security mission scenarios.

<u>OBSERVATION</u>: Surveillance and protection of national critical assets and populations as studied in this research, requires the definition of sensor system architectures able to detect people, as well as aerial, ground and maritime vehicles that could represent potential threats to the homeland. Determining a proper detection architecture for a particular critical asset of interest requires the consideration, combination and optimization of several detection systems that may interact with each other and with their environment, and that may vary in:

- Number
- Types
- Properties
- Positions in the architecture
- Performance
- Cost

and whose performance depends on the size of the item to detect and monitor, on the terrain features, and on the weather conditions.

MODELING AND SIMULATION – RESEARCH QUESTION: How can a portfolio of distributed detection system architectures be determined given the interactions of a variety of heterogeneous systems with each other and with their surrounding operational, geographic and climatic environments?

HYPOTHESIS 4: The determination of a portfolio of distributed detection system architectures will be facilitated by the definition of top level relationships capturing the interactions between the various elements of the homeland security mission, and allowing the comparison of the various system architectures with each other.

OBSERVATION: Detection architectures and detection systems may be characterized by a myriad of figures of merit, going from performance, cost, to reliability, survivability, flexibility, etc. However, for the problem at hand, what is of interest is:

1. How well the detection architecture scores at detecting people as well as aerial, ground, or maritime systems depending on the operational scenario
2. How much the implementation of the detection architecture costs to the decision maker

Therefore,

- The performance of the detection architecture, and
- The cost of the detection systems (with the exclusion of the cost of positioning those detection systems on the terrain, as well as any additional costs related to the purchase of complementary equipment, parcels of land, etc),

may be identified as the most appropriate measures of effectiveness for the detection architecture, as well as for the component detection systems.

OBSERVATION: The identification of an adequate portfolio of detection architectures given specific operational conditions, performance requirements and/or cost constraints requires the optimization of the detection systems composing the architecture, in terms of:

- Number
- Types

- Properties
- Positions in the detection architecture

depending on the operational situation, the terrain features and the climatic conditions.

OPTIMIZATION METHOD – RESEARCH QUESTION: How can the number, types, properties, and positions of heterogeneous systems, be optimized concurrently so as to define a distributed system architecture for a specific operational scenario, given performance requirements and/or cost constraints?

OBSERVATIONS:

- The problem of constructing a distributed detection system architecture from a pool of available types of sensors, with varying properties, that can be positioned at a specified set of positions on the terrain, in various operational, geographic and climatic environments is a highly discontinuous and non-linear problem that may present several local optima.
- Similarly, being able to detect a wide range of entities, whether it is people or aerial, ground, or maritime vehicles, in a given operational scenario involves several elements of different natures that need to be handled differently. This also presents discontinuous and non-linear properties that complicate the design and optimization of detection architectures.
- The problem involves two conflicting objectives – performance and cost – such that improving one of them may result in the degradation of the other.
- The problem requires the exploration and exploitation of the space of detection systems (in terms of number, types, properties and positions in the detection architecture) in order to devise an optimum detection architecture in a specific context of operations.
- The problem requires the determination of a portfolio of detection architectures adapted to a specific operational scenario according to performance requirements and/or cost constraints, rather than a single optimized detection architecture.

- The problem under study is composed of a large number of variables. This may be computationally expensive when trying to determine a portfolio of detection architectures since the objective function to be optimized needs to be evaluated several times.

HYPOTHESIS 4.1: Evolutionary optimization algorithms such as Genetic Algorithm (GA), or Particle Swarm Optimization (PSO), will provide a means to solve the multi-objective, discontinuous and non-linear optimization problem considered in this research, balance the tradeoff between exploration and exploitation, find a number of optimally efficient solutions rather than a single solution for the distributed system architectures in specific operational contexts, handle performance and/or cost constraints, and explore the search space more thoroughly with smaller numbers of objective function evaluations.

OBSERVATION: Evolutionary optimization algorithms such as GA or PSO have been shown to present convergence issues for highly dimensional, discontinuous, non-linear problems, due to the dependence of the algorithm parameters on the nature of the problem to which they are applied. Indeed, the efficiency of a GA or a PSO algorithm highly depends on the method employed to encode candidate solutions (binary encoding, many-character and real-valued encodings, tree encodings), on the operators, on the parameters settings, and on the particular convergence criterion.

OPTIMIZATION PARAMETER SETTINGS – RESEARCH QUESTION: How can we determine a set of optimization algorithm parameters adapted to the design of distributed detection system architectures to ensure good convergence properties and the adequacy of the resulting solutions?

HYPOTHESIS 4.2: Applying the GA or PSO to a simpler analytical test problem (whose solution is known) presenting similar discontinuous, non-linear, and dimensional properties as the original problem, and varying the algorithm parameters will provide a way to analyze the sensitivity of the solution to the algorithm parameter settings and combinations, and determine the set of algorithm parameter values that provides the most accurate solution for the test problem. Then, using this resulting set of algorithm parameter values on the original problem is assumed to ensure the convergence of the optimization algorithm to efficient distributed detection system architecture solutions.

OBSERVATION: Evolutionary optimization algorithms such as GA and PSO have been shown to yield solutions that may not always be reproducible for large dimensions, discontinuous, non-linear problems such as the design and optimization of distributed detection system architectures (DODA)

SOLUTIONS BENCHMARKING AND ACCURACY CHECKING – RESEARCH QUESTION: How can we check the accuracy of the solutions provided by the optimization algorithms when applied to the DODA problem?

HYPOTHESIS 4.3: Developing a heuristic recursive optimization scheme based on simple performance, cost and geometrical positioning rules will enable benchmarking and checking the accuracy of the detection architecture solutions provided by the evolutionary optimization approach.

OBSERVATION: The surveillance of geographic borders and critical assets using limited sensor capability has always been a challenging task in many homeland security applications. Geographic borders may be very long and may go through isolated areas that are sometimes uninhabitated. As for critical assets, they may be large and numerous and may be located in highly populated areas. As a result, it is virtually impossible to secure each and every mile of border around the country, and each and every critical infrastructure inside the country. Most often, a compromise must be made between the percentage of asset covered by surveillance systems and the induced cost. However, governments may be willing to invest differently into securing their borders and critical assets. Once potential detection architecture solutions have been obtained for a specific customer need, the goal is to verify their predicted performance in an actual operational context and to potentially complement the architecture by additional mobile or fixed systems so as to increase its operational effectiveness.

SOLUTIONS ANALYSIS – RESEARCH QUESTION 5: How can we rapidly, quantitatively, and efficiently assess the operational effectiveness of the portfolio of distributed detection system architectures obtained though evolutionary optimization?

HYPOTHESIS 5: A flexible agent-based and physics-based framework will allow creating a simple model so as to both rapidly, quantitatively, and efficiently evaluate the operational effectiveness of the portfolio of distributed system architectures and potentially complement the architecture by additional mobile or fixed sensor systems so as to increase its operational effectiveness.

OBSERVATION: The focus of the proposed study is the development of a structured, valid, defensible, adaptive and practical methodology facilitating multi-criteria decision making processes for the protection of critical assets and populations, in the context of homeland security. On the one hand, the methodology is intended to facilitate the quantitative assessment of the operational, economic and technology potential of detection architecture solutions, with respect to capability-level measures of effectiveness. On the other hand, the methodology is meant to allow a traceable analysis of the design and optimization factors enabling the decision maker to explore the design space and assess the relative sensitivity of tradeoffs at all levels of the hierarchy. This involves being able to assess the impacts of changes in the structure of the detection architecture on its performance and cost, when the number of detection systems, their types, properties, and/or positions on the terrain are modified.

WHAT-IF ANALYSIS – RESEARCH QUESTION 6: How can we assess the impacts of changes in the structure of the distributed system architecture on its performance and cost?

HYPOTHESIS 6: The flexible agent-based and physics-based framework will both allow the decision maker to play “what-if” scenarios, thus exploring the impact of varying the structure of the distributed system architecture on the performance and cost metrics, and support the development of a quantitative, transparent, adaptive and practical methodology, while ensuring the traceability and the accuracy/validity of the definition of distributed system architecture solutions for a specific homeland security mission.

3.3. Proposed Methodology

To truly understand the nature of the detection architecture, assess its capabilities, and capture the key tradeoffs between detection platforms, items of interest, critical assets, and operational environment, it is necessary to create sufficiently accurate representations in a modeling and simulation environment, where any number of notional "what-if" scenarios can be played. However, before any modeling effort can be undertaken, it is vital to recognize that the analyst is ultimately attempting to represent a wide variety of existing sensor systems (such as various types of radars, and various categories of cameras) as well as various items of interest identified as being relevant to the problem (such as general aviation aircraft, Unmanned Aerial Vehicles (UAVs), motorized gliders, ultralights, individual persons, groups of people, land vehicles, marine vessels, etc). At the same time, the analyst seeks to have the flexibility to generate notional detection systems representative of future advanced concepts. Similarly, it is highly desirable to have the freedom to consider notional threats that have not yet been conceived. This is also applicable to the representation of a wide range of critical assets (such as air, land, and maritime borders, airports, governmental assets, nuclear power plants, electric facilities, dams, commercial centers, chemical industries, refineries, national monuments and icons, etc), located in a variety of operational environments (i.e. various combinations of geographic and climatic conditions).

The analyst is thus challenged both by the need to characterize a wide range of existing elements and the incentive to generate new notional ones. Hence, a structured, yet flexible, characterization scheme must be adopted. Such a scheme is built on the concept of *parametric representations*, whereby a set of parameters of interest with their respective domain of allowable values is identified and used to generate different configurations of a given system, or different systems altogether, depending on the application. Such an approach has been successfully applied to the exploration of the design space of a variety of aerospace systems [430], as well as to the generation of operational scenarios [431]. This approach is also implicit in some stochastic optimization schemes which have also been successfully used in various defense and aerospace applications, such as genetic algorithms previously applied to aircraft design and optimization problems [432], [433], [434], [435].

To facilitate the generation of parametric representations, it is desirable to use a structured approach whereby the detection system-of-system is progressively decomposed into components with increasing levels of details, both physically and functionally. The resulting elements may then be adequately grouped, thus revealing sets of common parameters. In turn, once its various elements have been parametrically represented in a modeling environment, the initial system-of-system may be subsequently recomposed, or synthesized. Top-level capabilities, resulting from lower level representations and interactions, may finally be revealed. It is evident that this approach to parametric representations applies the fundamental systems engineering concepts of decomposition and synthesis, both at the physical and functional levels, as well as the concept of hierarchical decomposition [107], [436], as shown in Figure 11.

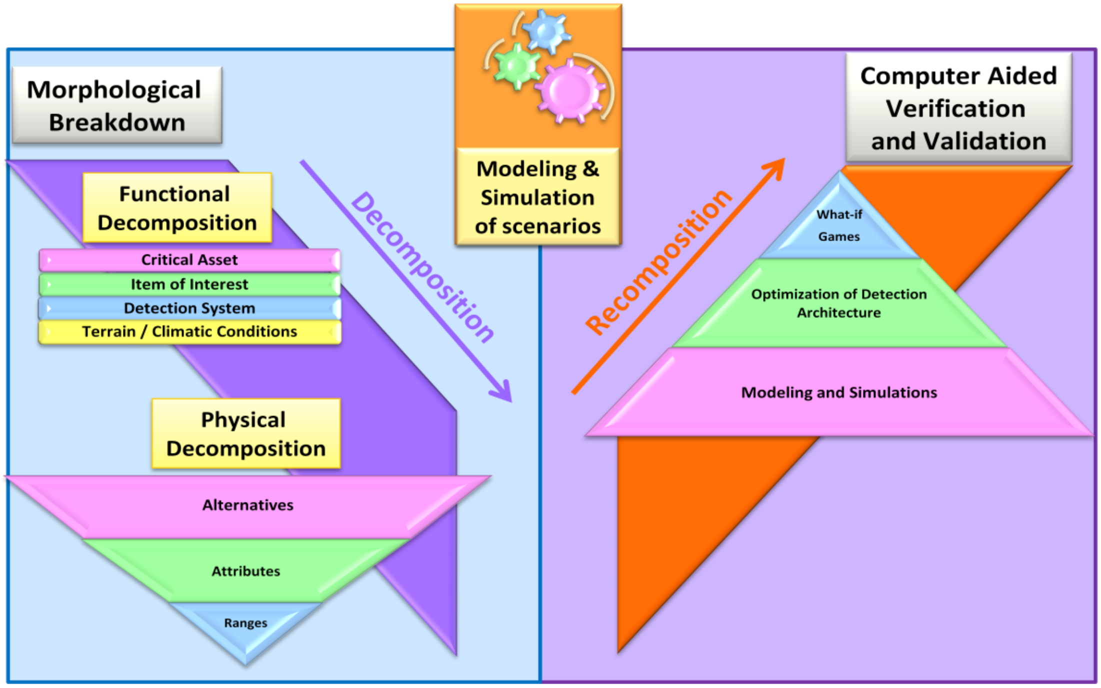


Figure 11: Decomposition/Recomposition Process Flowchart

In order to facilitate the Modeling, Simulation, and Optimization of Distributed Detection system Architectures, a seven-step methodology, named M-SODDA, is proposed. This methodology aims at discovering coverage- and cost-efficient combinations and geometrical configurations of fixed and mobile detection systems distributed over large terrains able to adequately detect items of interest evolving in a specific geographic and climatic environment. The seven steps of the M-SODDA methodology, as well as the general elements involved in each step, are displayed in Figure 12.

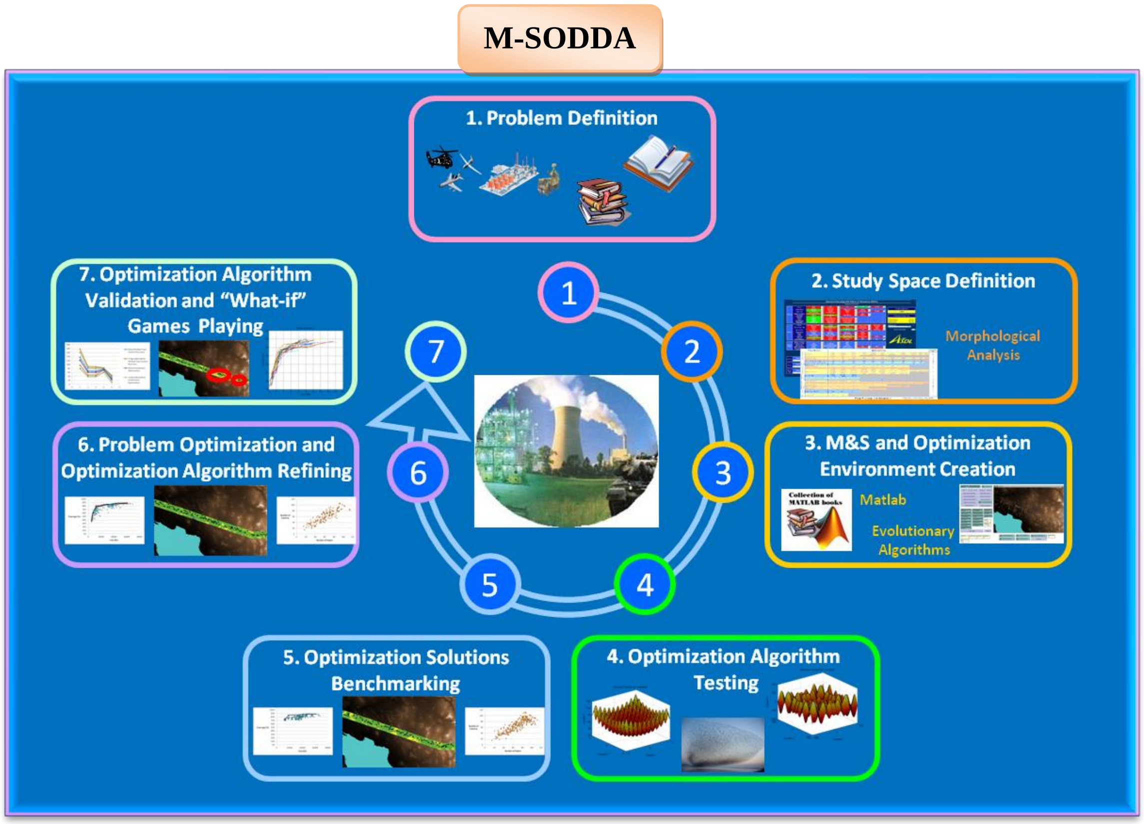


Figure 12: Proposed Methodology for the Multidisciplinary Design, Modeling, Simulation and Optimization of Architectures of Fixed and Mobile Detection Systems Distributed Over Large Areas for Surveillance and Protection Missions in Homeland Security (M-SODDA)

The first step of the M-SODDA methodology has already been addressed in Chapter I and Chapter II where the main motivation for this work was presented and the associated useful theory and literature was discussed.

The second step of the M-SODDA methodology features a detailed demonstration of the use and combination of enabling methods and tools for decomposing the DODA problem into its main elements and for reformulating the problem into operationally relevant terms. These methods and tools include *Functional Decomposition* from Systems Engineering, *Morphological Analysis*, *Interactions and Consistency Analysis*, *Heuristic Optimization*, *Evolutionary Optimization*, and *Sensitivity Analysis* among others. More precisely, the second step of the M-SODDA methodology consists of:

- The decomposition of the problem into its main elements, sub-elements, alternatives, attributes, and relevant representations of the various types of attributes indentified.
- The analysis of the compatibilities between the various elements, sub-elements, alternatives, attributes, and attributes representations, to determine consistent operational scenarios and eliminate infeasible or non-probable situations.

The M-SODDA methodology then focuses on a detailed description of a multidisciplinary strategy for the design and optimization of detection architectures in terms of detection performance and cost. More precisely, the third step uses the information gathered in the previous step to create a framework for the modeling, simulation, and optimization of detection architecture solutions for various potential operational scenarios of interest. The fourth, fifth, and sixth steps of the M-SODDA methodology then develop a structured and transparent process for the optimization of distributed detection system architectures in specific operational contexts, and for testing the accuracy of the resulting solutions. Finally, the seventh step of the M-SODDA methodology features guideline principles for checking the accuracy of the detection architecture solutions obtained in the previous steps and for customizing their properties by performing various kinds of sensitivity analyses and "what-if" games. More precisely, the last five steps of the M-SODDA methodology consists of:

- The modeling of the elements involved in the study, namely critical areas (assets or borders), items of interest, detection systems, and geographic and climatic conditions, using physics-based and agent-based models.
- The investigation of optimization approaches adapted to the mixed, multi-criteria, multi-dimensional, highly discontinuous problem of DODA.
- The determination of a set of optimization algorithm parameters adapted to the problem of DODA by using testing functions presenting similar characteristics as the original optimization problem.
- The down-selection and the refining of an optimization approach for obtaining distributed detection system architecture solutions for the problem of DODA.

- The development of a heuristic optimization approach to provide benchmark detection architectures for the solutions provided by the refined optimization algorithm.
- The optimization of the structure and the composition of detection architectures able to perform the original detection and surveillance mission in the operational situation considered, using the modified optimization algorithm. The detection architectures are initially statically optimized, meaning that they are composed of fixed sensor systems distributed over the area under consideration, at locations determined to be promising in terms of detection performance.
- The investigation of the accuracy of the solutions provided by the refined optimization approach applied to the original optimization problem using the benchmark detection architectures obtained from the heuristic optimization algorithm.
- The analysis of the performance of fixed Pareto efficient detection architectures provided by the modified optimization approach and their enhancement with mobile detection systems in regions of the theater of operations lacking detection coverage.
- The analysis of the sensitivities of the performance and cost of the complete Pareto efficient detection architectures, featuring both fixed and mobile detection systems, to changes in their structures and their compositions, to obtain customized detection architecture solutions. This may be performed directly by the end user in the modeling and simulation framework developed as part of this work, by playing any kind of "what-if" games on the detection architectures.

In this work, two homeland security applications will be considered to demonstrate the M-SODDA methodology. The first example concerns the homeland security mission of Critical Assets Protection (CAP). It refers to the protection of those assets that are so vital to the stability and the normal operation of a nation that their disruption or destruction are expected to have a debilitating impact across economic, social, health, and other dimensions. This application will be used to demonstrate the first two steps of the proposed methodology. The second example concerns the homeland security mission of

Customs and Border Protection (CBP). It refers to the surveillance of large terrains and geographic borders against potential smugglers, drug dealers, illegal border crossings, and malevolent entities who are trying to enter a country to exploit or harm its people or infrastructures. It focuses on the southwest land border between the state of Arizona in the United States of America and the state of Sonora in Mexico. It implies the design, modeling, simulation, and optimization of multispectral three-dimensional detection architectures for border protection and intrusion detection at the Arizona-Sonora border and will be used to demonstrate the last five steps of the proposed methodology.

CHAPTER IV

IMPLEMENTATION – PROBLEM DECOMPOSITION AND ANALYSIS

4.1. <u>Morphological Analysis of a Proof-of-Concept Scenario</u>

This section addresses the parametric analysis research question along with the corresponding problem decomposition, morphological analysis, morphological analysis revisited, cross-consistency assessments, and cross-consistency assessments revisited research questions, and serves as a test to the corresponding hypotheses.

A convenient way to structure system's decomposition for parametric representations, and the subsequent synthesis of the detection SoS, is through Morphological Analysis (MA). This approach uses a morphological matrix to document how a system of interest is decomposed into main element classes, and enumerates possible alternatives for each element class. A Cross Consistency Matrix (CCM) documents relational data between element alternatives, thus establishing the combinatorial logic that drives the synthesis of element alternatives into a number of internally consistent system's configurations. This research incorporates two important improvements to the traditional formulation of MA.

First, it explicitly incorporates a multi-level approach accommodating any successive decomposition steps that may be required, thus more closely following the conceptual formulation of the *systems engineering "Vee"* [96]. Hence, in a first step, the problem under consideration is decomposed into main parameters of importance, namely the critical area to be protected (asset or border), the detection system(s), the terrain or geographic environment, the potential "threats" to the area to protect or items of interest, and the climatic conditions under which a scenario can take place. Several alternatives are then brainstormed for each parameter and regrouped in a High-Level or Operational-Level Morphological Matrix (HLMM). Then, the problem is further decomposed, and relevant attributes for each element of the problem are identified. In a last step, appropriate ranges of values (for continuous variables), or appropriate discrete values (for discrete variables), and appropriate ways of representing qualitative variables are determined for each attribute. This allows the creation of a Sub-Level or System-Level Morphological Matrix (SLMM).

Second, this research builds on previous efforts [93] to assess the consistency of the information contained in the HLMM and in the SLMM, and encodes relational data with higher resolution scales to capture more complex interactions. More precisely, it implements a probabilistic scale instead of the traditional MA binary scale (0 for incompatibility and 1 for compatibility) to introduce some nuance in the cross-consistency assessment. This method allows specifying more accurately the degree of likelihood that two elements can coexist in a given operational scenario. The consistency of the information contained in the Morphological Matrices is finally assessed at both the operational level and the system level. This enables the creation of two Cross-Consistency Matrices (Operational-Level and System-Level: HLCCM and SLCCM) and paves the way for the modeling and simulation of candidate scenarios defined from the Sub-Level Morphological Matrices.

4.1.1. High-Level Decomposition

In this adapted version of MA, the decomposition process begins at the highest level where the problem construct is decomposed into corresponding functional and physical parts. As has been mentioned, there are three primary component types in this application, namely the asset to be protected, the detection systems, and the items of interest "threatening" the critical area under study. Let us consider the example of a Critical Assets Protection (CAP) mission which refers to the protection of those assets that are so vital to the stability and normal operation of a nation that their disruption or destruction are expected to have a debilitating impact across economic, social, health, and other dimensions [9],[437],[438]. Such assets can be physical infrastructures, land and maritime borders, as well as virtual aerial boundaries. Though threats to critical assets can be conceived to be of various forms [18], those related to low or very low altitude aerial systems have been recognized to be of particular interest by some governments [21]. As a result, the development of customized detection architectures for airborne threats to critical assets has been identified as a major priority by the defense community [10]. Indeed, there are three primary threats to the Air Domain: to and from aircraft, to the Aviation Transportation System Infrastructure, and from hostile exploitation of cargo.

1. The threats to and from aircraft are composed of large passenger aircraft, large all-cargo aircraft, small aircraft, non-traditional aircraft such as UAVs, ultra-lights, gliders or aerial-application aircraft. These categories might be susceptible to carrying explosives, weapons of mass destruction, to spying on a particular critical asset, to smuggling of terrorists and instruments of terror, or be subject to hijacking as well as be the prey of Man Portable Air Defense Systems (MANPADS).

    a. Large passenger aircraft have been at greatest risk to terrorism due to their potential to inflict catastrophic damage and their likelihood to disrupt the Aviation Transportation System. Whether passengers and aircraft are used as targets or aircraft are used as weapons, the goal is always to inflict as much damage as possible, aiming at the greatest death toll.

    b. Large all-cargo aircraft might be used when more attractive targets such as large passenger aircraft are absent. If terrorists adapt their tactics this way, large all-cargo aircraft may be more attractive as weapons, such as through hijacking to attack ground targets, or as conveyance means.

    c. For their part, small aircraft might be used in high numbers as weapons to destroy a critical asset or portion of infrastructure. The most serious threat involving small aircraft is the transport of weapons of mass destruction or related materiel, the transport and release of chemical or biological agents, as well as spying such as videotaping or taking pictures of a critical asset in order to plan for a large scale attack.

    d. Finally, non-traditional aircraft might be employed as weapons, as a means to disseminate weapons of mass destruction or to release chemical or biological agents, or as a means of espionage to prepare a future attack on a critical asset. For example, terrorists might use non-traditional aircraft for missions of limited range, requiring limited accuracy or having a specific target and small asset in mind, or simply to videotape or take pictures of more or less accessible asset(s) to plan for eventual large scale attack(s).

2. The threats to the Aviation Transportation System Infrastructure are relatively few due to the relatively low public profile of Aviation Navigation Services such as Air Traffic Control facilities and systems, the robustness and resilience of these systems thanks to many layers of redundancies, and the Nation's likely capacity to recover rapidly and thus limit psychological or economic impact of any attack. However, despite all these measures to deter terrorists from attacking the Aviation Transportation System Infrastructure, terrorists might target passenger concentration at commercial airports, recycling tactics from many years ago. Terrorists might also target multi-use airports such as those combining commercial and military operations, or commercial and general aviation operations.

In the context of the current CAP application, the different elements involved fulfill corresponding top-level roles/functions, namely "be protected", "detect" and "threaten the protection mission". Additionally, the operational environment is recognized as a key driver impacting performance at all levels, and is represented as an additional element of the system, both in terms of terrain/geographic conditions and climatic conditions.

Each of these elements is then further decomposed into sub-elements, as shown in Figure 13, effectively implementing a new level in the hierarchical structure of the problem construct. The definition and determination of sub-elements was conducted through brainstorming and literature search [11],[19],[140],[143],[150],[151],[178],[439],[440], and was complemented by multiple iterations with subject matter experts. Finally, for each sub-element, a list of possible alternatives is identified. A balance between exhaustiveness and relevance to the current application is key in providing an alternatives set that is neither unmanageable nor incomplete. Similarly, alternatives were defined through brainstorming and literature search [4],[12],[14],[18],[441],[442],[443],[444], and complemented by multiple iterations with subject matter experts. The hierarchical breakdown of top-level elements, sub-elements, and their corresponding alternatives is documented in the High Level

Morphological Matrix (HLMM). This is shown in Figure 13 for the homeland airspace protection (CAP) mission.

ELEMENT	SUB-ELEMENT	ALTERNATIVES				
CRITICAL ASSET	*Type*	Agriculture and Food	Water	Public Health	Emergency Services	Defense Industrial Base
		Information, Telecommunications and Technologies	Energy	Transportation	Banking and Finance	Chemicals and Hazardous Materials
		Postal and Shipping/Civil Administration	Space and Research	Public & Legal Order and Safety	National Monuments and Icons	Nuclear Power Plants
		Dams	Government Facilities	Military Assets	Commercial Key Assets	
	Scope of Impact	Local	Regional	National	International	
	Degree of Loss or Severity	Immediate Impact	Impact in 24 to 48 hours	Impact in One Week	Other	
	Effect of Time	None	Minimal	Moderate	Major	
	Impact Assessment	Public	Economic	Environment	Political effects	Psychological Effects
		Interdependency				
TERRAIN	*Mountain*	Isolated Mountain	Chain of Mountains			
	Body of Water	River	Lake	Sea/Ocean/Coast		
	Forest	Very small	Small	Medium	Large	Very large
	Urban Environment	Cluster of houses	Town	City	City + Neighborhoods	Metropolis
CLIMATIC CONDITIONS	*Atmosphere*	Tropical	Mid-latitude Summer	Mid-latitude Winter	Subarctic Summer	Subarctic Winter
	Aerosol State	Clear	Hazy			
ITEM OF INTEREST	*Type*	General Aviation Aircraft	Crop Duster	Ultralight	Motorized Glider	Helicopter
		Unmanned Air Vehicle	Air to Surface Missile	Cruise Missile		

Figure 13: High Level Morphological Matrix of the CAP Mission

SENSOR	*Active Radar*	Classic Radar + Mechanical Scan	Classic Radar + Mechanical Scan + Track While Scan	Classic Radar + Electronic Scan	Classic Radar + Electronic Scan + Track While Scan	UHF Radar + Mechanical Scan
		UHF Radar + Mechanical Scan + Track While Scan	UHF Radar + Electronic Scan	UHF Radar + Electronic Scan + Track While Scan		
	Passive Radar	3 Tx[1] + 1 Rx[2] + CU[3]	1 Tx + 3 Rx + CU	Multiple Tx + 1 Rx + CU	1 Tx + Multiple Rx + CU	
	Optronic System	Infrared 1	Infrared 1 Bi-field	Infrared 2	Infrared 2 Bi-field	Infrared 3
		Infrared 3 Bi-field				

Figure 13: High Level Morphological Matrix of the CAP Mission

The next part of the decomposition deals with the definition of sets of attributes and their associated ranges of values with which *parametric representations* of system elements can be effectively made with a modeling and simulation capability in mind. The selection of said attributes and value ranges should be such that they can adequately capture the spectrum of alternatives prescribed for each "Element" in the HLMM, and should furthermore allow for additional parametric definitions of other alternatives not considered therein. It is important to note that attributes may be continuous or discrete, and that value ranges are therefore defined accordingly. This phase of the system decomposition effort is certainly not trivial. Indeed, the objective is to capture features of relevance across a heterogeneous set of alternatives through a small number of attributes. This way, the analysis has an acceptable fidelity but remains manageable. Moreover, identifying what these attributes are can be cumbersome and may require extensive literature search and iterations with subject matter experts. In some instances, however, relevant attributes are readily identified, for instance as the basic design parameters of a radar platform. The result of this process, namely the definition of attributes and value

[1] Tx is the abbreviation for Transmitter

[2] Rx is the abbreviation for Receiver

[3] CU is the abbreviation for Central Unit

[4] Cases where the detection coverage is between 0% and 1% are highly improbable. There will always be at least one sensor system in the architecture such that its coverage will always be strictly larger than 1% even

ranges of each sub-element, is documented in the Sub Level Morphological Matrix (SLMM).

4.1.2. Sub-Level Decomposition

4.1.2.1. Defended Asset Element

In the homeland airspace protection mission, a salient feature for the parametric representation of critical assets is their relative importance to the decision maker so that an appropriate level of protection required can be specified in every particular case. Consequently, the problem is best approached by giving the decision maker the opportunity to define a level of protection to be achieved with the detection SoS. In this context, the focus being only on the detection of potential threats to specific critical assets, a relevant attribute to be considered is the required probability of detection of an airborne threat, within a given distance from the defended asset. Moreover, the level of protection might not only depend on the specific asset in mind, but also be a function of the performance of protection systems (especially effectors) that may already be in place. In order to capture the efficiency of said known or unknown systems, field experts usually consider the time required to deal with a potential harmful threat with such systems. This gives a distance before which a potential threat has to be detected and identified to be efficiently handled. This distance can be represented as a sphere of protection around the defended asset, or as a cylinder if it is associated with a notion of limit altitude of detection. The attributes and ranges in the SLMM for the "critical asset" in the CAP mission are provided in Table 1. This process of defining relevant attributes for the defended asset obviously implies several iterations with experts in the field so as to determine a set of adequate attributes capturing the whole extent of the problem while still yielding a manageable design space.

Table 1: Attributes and Ranges for the Critical Asset in the CAP Mission

CRITICAL ASSET	ATTRIBUTE	UNIT	MINIMUM VALUE	MAXIMUM VALUE
	Probability of Detection Required for	%	0	100

	the SoS before Limit Distance			
	Time Required for Effectors to act from Defended Asset	*min*	0	30

4.1.2.2. Aerial System Element

The definition of attributes for the "item of interest (IoI)" element necessitates taking into account requirements for the modeling of detection systems, essentially the calculation of their performance metrics depending on the "item of interest" attributes. Therefore, the wide variety of potential items of interest can be captured into five attributes for the CAP mission. The attributes and ranges in the SLMM for the "item of interest" are provided in Table 2. Once again, the definition of relevant attributes for the "item of interest" involves multiple iterations with experts in the field so as to determine a set of adequate attributes capturing the whole extent of the problem while still yielding a manageable design space.

Table 2: Attributes and Ranges for the Item of Interest in the CAP Mission

	ATTRIBUTE	UNIT	MINIMUM VALUE	MAXIMUM VALUE
ITEM OF INTEREST	*Speed*	*m/s*	1	1600
	Altitude	*m*	0	12000
	Range	*m*	0	20000
	Heading	°	0	360
	Radar Cross Section (RCS)	dm^2	0	10000

4.1.2.3. Geography and Climate Elements

A similar approach can be adopted to define appropriate attributes for the terrain and the climatic conditions. A wide spectrum of terrain features is considered, such as mountains, bodies of water, forests, and urban environments. Nevertheless, in a scenario simulation, the characteristics of primary operational importance are the general size and shape of each feature and its distance to the critical asset.

As for the climatic conditions, they essentially influence the performance of the optronic detection systems (cameras). Indeed, radars, whether active or passive, are relatively insensitive to the presence of clouds, rain, snow and fog through which they can "see" without any restrictions. However, optronic detection systems are highly sensitive to the amount of aerosols in the air. A detailed private study from a foreign government industry on the interactions between the performance of optronic detection systems and the numerous atmospheric properties shows that two weather-related attributes are sufficient. These are cloud ceiling for detection (representing the altitude of eventual cloud coverage above the theater of operations) and visibility (representing the amount of aerosols in the air). The attributes and ranges in the SLMM for the terrain and the climatic conditions are provided in Table 3 and Table 4 respectively.

Table 3: Attributes and Ranges for the Geography Element in the CAP Mission

TERRAIN	*Type*	Mountainous Environment	Isolated Peak
			Chain
		Aqueous Environment	River
			Lake
			Coast
		Forested Environment	Small
			Medium
			Large
		Urban Environment	Cluster of Houses
			Town
			City
			City + Neighborhoods
			Metropolis

	ATTRIBUTE	UNIT	MIN VALUE	MAX VALUE
TERRAIN	*Distance from Defended Asset*	*m*	0	10000
	Surface Covered	km^2	0	12000
	Average Height	*m*	0	9000
	Angular Location from Defended Asset	°	0	360
	Width (for River)	*m*	0	500

Table 4: Attributes and Ranges for the Climate Element in the CAP Mission

	ATTRIBUTE	UNIT	MIN VALUE	MAX VALUE
CLIMATIC CONDITIONS	*Clouds Ceiling for Detection*	*m*	50	12000
	Visibility	*km*	0	50

4.1.2.4. Detection System Elements

As mentioned previously, the attributes characterizing the detection systems of interest, namely active radars and visible or infrared cameras, fall out directly from the intrinsic design parameters of those systems. The attributes and ranges in the SLMM for the detection systems element are provided in Table 5 and Table 6.

Table 5: Attributes and Ranges for the Active Radar Element in the CAP Mission

ACTIVE RADAR	*Type*	Classic		
		UHF		
	Scanning Scheme	Mechanical		
		Electronic		
	Waves Propagation Scheme for UHF Radar	LOS		
		NLOS		
	ATTRIBUTE	**UNIT**	**MIN VALUE**	**MAX VALUE**
	Peak Emitted Power	*W*	10	10000
	Frequency of UHF Radar	*MHz*	300	1000
	Frequency of Classic Radar	*GHz*	1	10
	Losses + Noise Factor	*dB*	3	21
	Bandwidth	*MHz*	1	300
	Elevation Coverage	°	0	60
	Azimuth Coverage	°	0	360
	Antenna Length	*# of wavelength*	1	50
	Antenna Height	*# of wavelength*	1	50
	Speed of Scan if Applicable	*Rotation per…s*	1	15

Table 6: Attributes and Ranges for the Optronic System Element in the CAP Mission

OPTRONIC SYSTEM	*Type*	1		
		2		
		3		
	Bi-field	Yes		
		No		
	ATTRIBUTE	**UNIT**	**MIN VALUE**	**MAXVALUE**
	Wavelength Infrared 1	*μm*	10	10
	Wavelength Infrared 2	*μm*	3	5
	Wavelength Infrared 3	*μm*	8	12
	ARRAY SIZE			
	Number of Horizontal Pixels	-	480	480
	Number of Vertical Pixels	-	640	640
	Focal Dimension	*cm*	1	100
	Optic Diameter	*cm*	3	15
	OPTICAL SYSTEM			
	Detection Range	*km*	1	20
	Sensibility (NedT)	*mK*	1	150
	Integration Time Infrared 1	*ms*	40	40
	Integration Time Infrared 2	*ms*	20	20
	Integration Time Infrared 3	*ms*	5	5

The definition of a parameter space in the SLMM allows the analyst to create parametric representations for each of the element alternatives prescribed in the HLMM. All alternatives for a given element are represented by a common vector of parameters, and are distinguished from each other by means of the specific values selected. This provides the analyst with the flexibility of creating an operational scenario to be modeled in two ways. One is to use the HLMM, and directly select prescribed alternatives for which parametric representations have been provided. The other is to use the SLMM and dial in one by one the attribute values for all elements, hence departing from prescribed alternatives and formulating notional ones. A combination of the two may also be used, for instance by selecting alternatives from the HLMM as a starting point, and then implementing small perturbations by altering parameter values. A summary of the process followed to obtain the Morphological Matrices is provided in Figure 14.

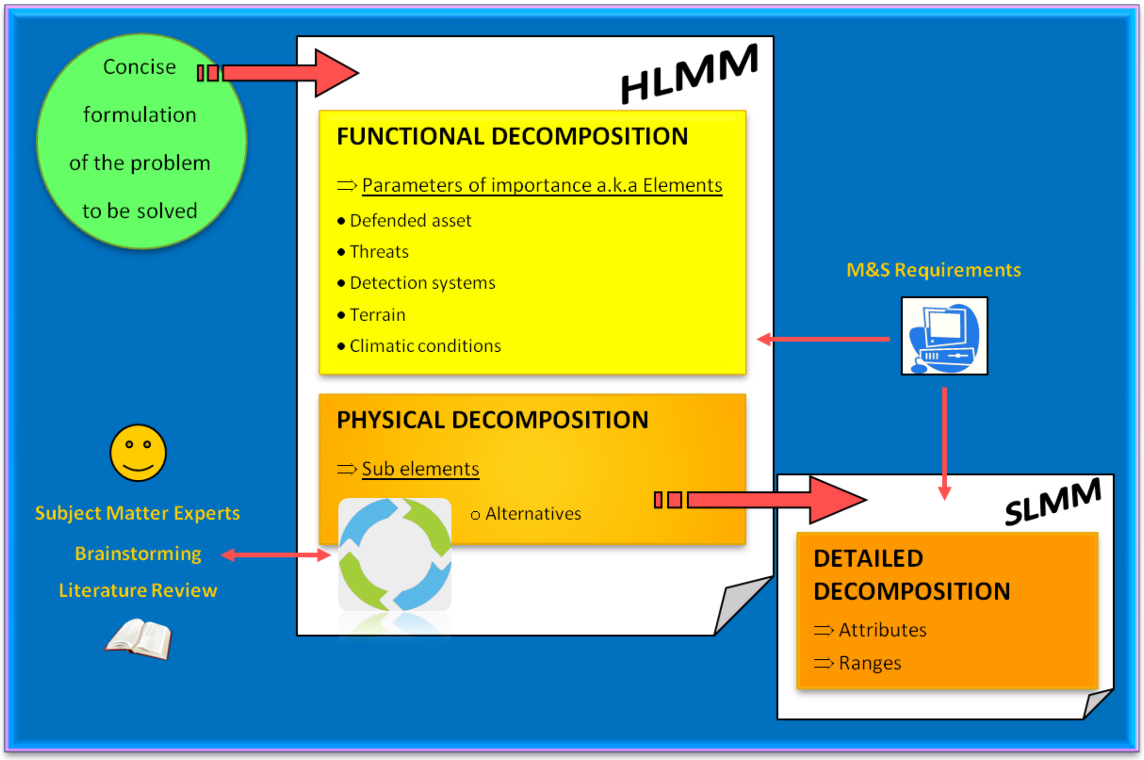


Figure 14: Morphological Decomposition Process Flowchart

4.2. Cross-Consistency Assessments

A typical morphological matrix can contain far too many configurations of alternatives to be inspected by hand. For instance, assuming that the alternatives in the High Level Morphological Matrix in Figure 13 are all compatible with each other, the total number of configurations or scenarios adds up to $3.48*10^9$! Thus, the next step in the proposed approach is to examine the internal relationships between parameters in the Morphological Matrix and to "reduce" the number of combinations by weeding out all mutually contradictory conditions. This is achieved through a cross-consistency assessment. Each "alternative" in the High Level Morphological Matrix is compared, in a pair-wise manner, to all of the others, much like a cross-impact matrix. For most problems, with each pair-wise relation, a judgment is made as to whether – or to what extent – the pair can co-exist, i.e. whether it is a compatible or an incompatible relationship. The results of cross-consistency assessments between high level alternatives are graphically depicted in a High Level Cross-Consistency Matrix (HLCCM).

4.2.1.1. Probabilistic Cross-Consistency Assessments

For the problem under consideration, a simple assessment of compatibilities is not enough and is actually not appropriate. For instance, in the cases:

- of the "item(s) of interest" with respect to the terrain or
- of the critical asset with respect to the "item(s) of interest",

the question "Is this alternative or concept compatible with this other alternative?" does not have a simple "yes" or "no" answer. Most of the time, the answer could be "it depends" or "well, not really but the alternatives are not strictly incompatible either".

To solve this ambiguity, one can describe relative consistencies in terms of probability or likelihood that two alternatives or concepts can coexist in reality, i.e. in an operational scenario, or even in a notional scenario.

To perform this task, the following scale is proposed:

0% Compatibility
- **0**: the alternatives are strictly (always) **incompatible** as part of their very nature

Low Compatibility
- **0.2**: it is **very unlikely** to find these two alternatives coexist in reality, and thus in a scenario (based on experience or on empirical grounds)
- **0.4**: it is **unlikely** to find these two alternatives coexist in reality, and thus in a scenario (based on experience or on empirical grounds)
- **0.6**: it is **likely** to find these two alternatives coexist in reality, and thus in a scenario (based on experience or on empirical grounds)

High Compatibility
- **0.8**: it is **very likely** to find these two alternatives coexist in reality, and thus in a scenario (based on experience or on empirical grounds) but they are not strictly compatible, i.e. they cannot coexist in all cases

100% Compatibility
- **1**: the alternatives are strictly (always) **compatible** as part of their very nature

4.2.1.2. Examples of Cross-Consistency Assessments

In the CAP application, each terrain alternative is compatible with all others and with itself, in the sense that one can define the same terrain feature more than once, with similar or different properties. For instance, the user can choose to define two isolated mountains in his/her scenario. Therefore, the diagonal portion of the High Level Cross-Consistency Matrix (HLCCM) for the terrain is composed of only 1s. . Similarly, each “item of interest” alternative is compatible with all others and with itself, in the sense that one can define the same “item of interest” more than once with similar or different properties. For instance, the user can choose to consider two general aviation aircraft in his/her scenario, yielding only 1s in the diagonal portion of the High Level Cross-Consistency Matrix for the “item of interest.”

Furthermore, there are no strict incompatibilities between “item of interest” alternatives and “critical asset” alternatives, meaning that there exists or will exist (maybe some time in the future) a possibility for a given "item of interest" to be a threat to a given critical asset, but that this threat is more or less likely to exist in real life. For example, a crop duster attacking a nuclear power plant would not make any significant damage and is thus assumed very unlikely. This example is displayed in red in Figure 15.

Critical Asset / Threat	Agriculture and Food	Water	Public Health	Emergency Services	National Monuments and Icons	Nuclear Power Plants	Dams	Government Facilities	Military Assets	Commercial Key Assets
General Aviation Aircraft	1	1	1	1	1	1	1	1	1	1
Crop Duster	1	1	1	0.3	1	0.2	0.2	0.2	0.2	1
Ultralight	1	1	0.2	0.2	1	1	1	1	1	1
Motorized Glider	1	1	0.3	0.5	1	1	1	1	1	1
UAV	0.4	0.4	0.4	0.2	1	1	0.6	1	1	1

Figure 15: Snapshot of a Cross-Portion of the High Level Cross-Consistency Matrix for the Critical Asset and the Item of Interest, in the CAP Mission

Similarly, most of the "item of interest" alternatives are strictly compatible with most of the terrain features. However, once again, some nuance can be introduced. For instance, a crop duster is very unlikely to be found in a mountainous environment or in an urban environment such as a medium or a big city. This case is depicted in red in Figure 16.

Terrain / Threat	Mountainous Environment	Cluster of Houses	Town	City	City and Neighborhoods	Metropolis
General Aviation Aircraft	1	1	1	1	1	1
Crop Duster	0.2	1	1	0.2	0.2	0.2
Ultralight	1	1	1	0.3	0.2	0.2
Motorized Glider	1	1	1	0.3	0.2	0.2
UAV	1	1	1	1	0.4	0.4

Figure 16: Snapshot of a Cross-Portion of the High Level Cross-Consistency Matrix for the Terrain Element and the Item of Interest, in the CAP Mission

As can be inferred from the previous discussion, determining high level compatibilities does not yield a satisfactory assessment since it seems that almost every alternative is compatible with every other. Indeed, apart from very isolated cases (such as those described above), the HLCCM is mostly composed of 1s. Therefore, in order to specify the likelihood assessments in the HLCCM, it is necessary to go down one level of definition and apply a similar process to the attributes and ranges of the SLMM so as to create a Sub-Level Cross-Consistency Matrix (SLCCM). Nevertheless, once again, a simple assessment of attributes' compatibility with one another is not enough and is actually not appropriate. Indeed, in most cases, two given attributes are neither strictly compatible nor strictly incompatible but their compatibility depends on their respective values.

To capture these dependencies, a 0, 1, 2 scale may be used in the creation of the Sub-Level Cross-Consistency Matrix:

- 0: the attributes compared are never compatible regardless of their respective values
- 1: the attributes compared are always compatible regardless of their respective values
- 2: the attributes compared can be compatible or not depending on their respective values

For example, the RCS of an "item of interest" depends on the type of radar considered, so this pair is given a 2, as depicted in red in Figure 17.

Threat / Active Radar	Average Speed (m/s)	Average Altitude (m)	Average Range (m)	Heading (°)	Radar Cross Section (dm^2)
Type	1	1	1	1	0.2
Peak Emitted Power (W)	1	1	1	1	1
Losses and Noise Factor (dB)	1	1	1	1	1
Bandwidth (Hz)	1	1	1	1	1
Elevation Coverage (°)	1	1	1	1	1
Azimuth Coverage (°)	1	1	1	1	1
Antenna Length (# wavelength)	1	1	1	1	1
Antenna Height (# wavelength)	1	1	1	1	1
Scanning Scheme	1	1	1	1	1
Speed of Scan if Applicable (rotation per s)	1	1	1	1	1

Figure 17: Snapshot of a Cross-Portion of the Sub Level Cross-Consistency Matrix for the Active Radar Element and the Item of Interest, in the CAP Mission

Moreover, it seems quite clear that the definition of the detection zone around the critical asset is highly dependent on the terrain. For instance, the radius of a sphere of protection around the critical asset, depending on the level of protection required and the time required for the effector systems to deal with a potential harmful aerial system is mostly determined by the terrain features surrounding the defended asset and their relative distance and location with respect to the latter. This translates into a cross-consistency assessment between attributes of the critical asset and attributes of the terrain yielding mostly 2s, as depicted in the red square in Figure 18 for the radius of the sphere of protection.

Critical Asset / Terrain	Probability of Detection Required for the Architecture Before Limit Distance (%)	Time Required for Effectors to act From Defended Asset (min)
Type	0.2	0.2
Distance From Critical Asset (m)	0.2	0.2
Surface Covered (m^2)	0.2	0.2
Average Height (m)	0.2	0.2
Average Location From Critical Asset (°)	0.2	0.2
Width (for Rivers)	0.2	0.2

Figure 18: Snapshot of a Cross-Portion of the Sub Level Cross-Consistency Matrix for the Terrain Element and the Critical Asset, in the CAP Mission

From this point on, pairs of attributes which were given a 2 are investigated further to determine for which of their values they are compatible and for which they are not. Several kinds of dependencies, leading a pair of attributes to be "not always compatible" as well as "not always incompatible", may be identified as part of this task. These include, but are not limited to:

- Mathematical dependency (e.g. speed and range for the "item of interest")
- Technological or state of the art dependency (e.g. number of horizontal pixels and number of vertical pixels constituting the detection array of an optronic system)
- Operational or scenario dependency (location of the transmitter/receiver antennae for passive radar systems and terrain features)

After having fully determined the "compatibilities" at the sub level, the "compatibilities" at the high level can be better specified. Moreover, the process consisting of going back up from the Sub-Level Cross-Consistency Matrix to the High Level Cross-Consistency Matrix, may help identify some situations or scenarios or combinations of alternatives in the High Level Morphological Matrix which are incompatible, although they appeared to be compatible, or more or less likely, in the first place. As such, the creation of the Cross-Consistency Matrices is really an iterative process, as is the creation of the Morphological Matrices, in the sense that their adequate creation requires several back-and-forth between accurate definition of the problem and M&S requirements. A summary of the process followed to obtain the Cross-Consistency Assessments is provided in Figure 19.

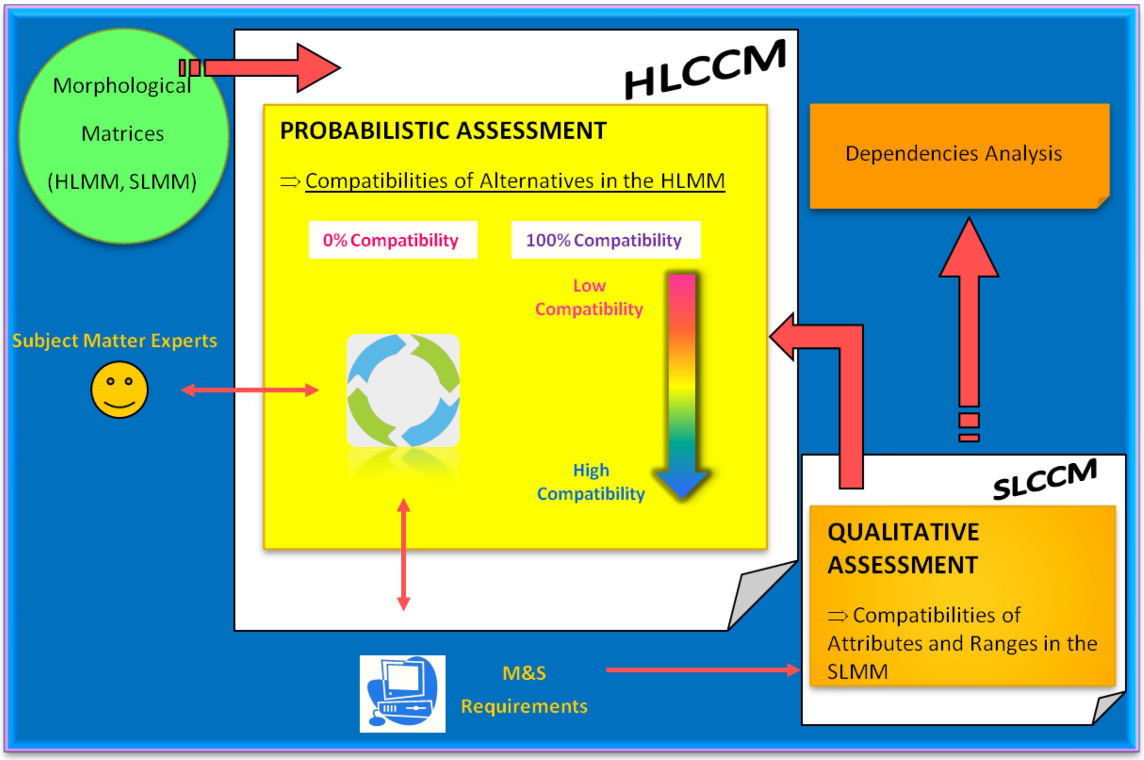


Figure 19: Cross-Consistency Assessments Process Flowchart

To conclude, Figure 20 depicts flowcharts of the morphological decomposition and of the cross-consistency assessment for the problem under consideration.

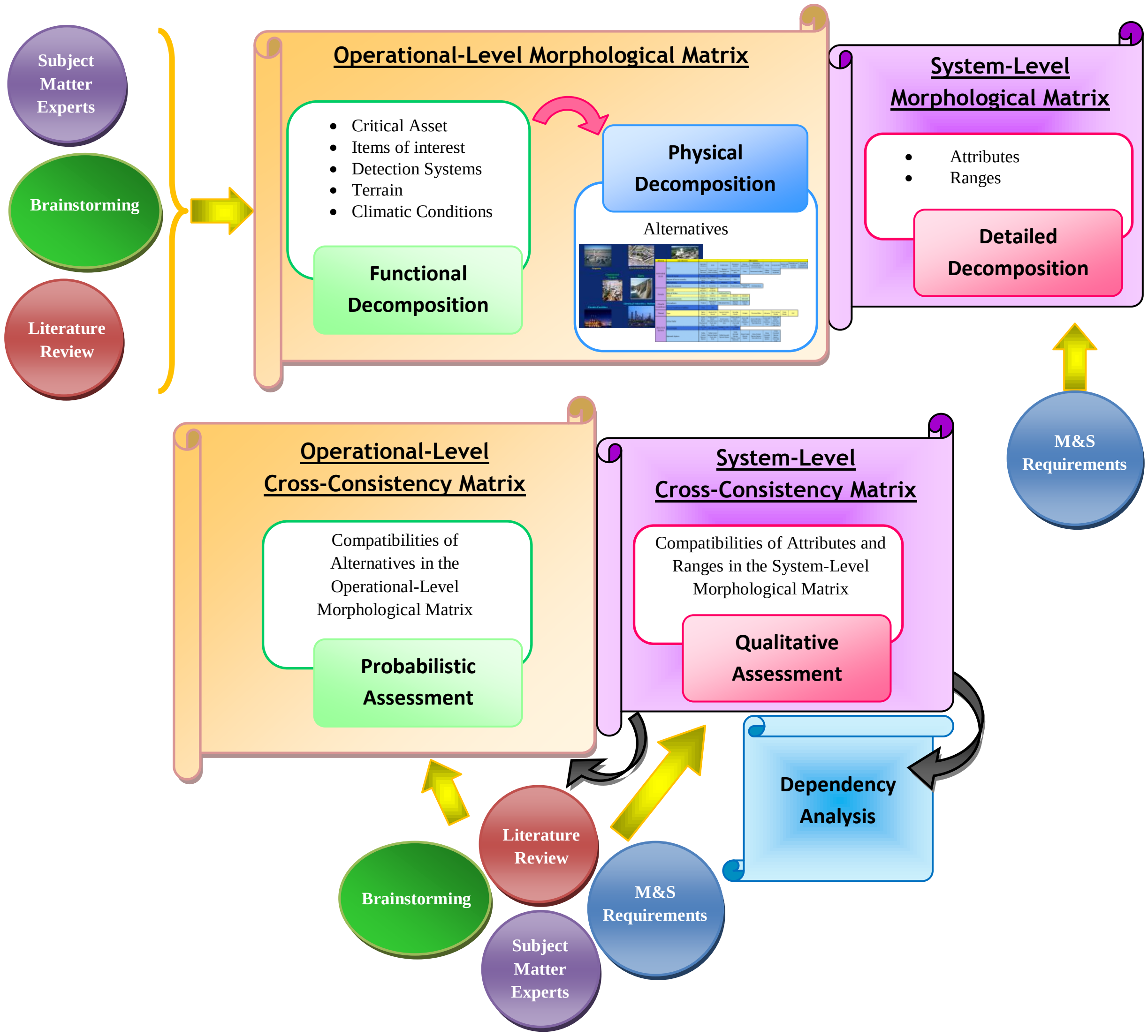


Figure 20: Flowcharts of the Morphological Decomposition and of the Cross-Consistency Assessment

4.3. <u>Concluding Remarks on Morphological Analysis and Cross-Consistency Assessments</u>

The sections above have demonstrated the construction of a structured, yet flexible characterization scheme whereby the homeland security problem of interest is progressively decomposed into its main elements, both physically and functionally to generate parametric representations. This was done by adequately regrouping different elements of the problem, thus revealing sets of common parameters.

First, a multi-level approach has been incorporated to the original Morphological Analysis (MA) method in order to determine a set of alternatives that best matches all levels of decomposition, and to accommodate any successive decomposition steps that may be required, thus more closely following the conceptual formulation of the systems engineering “Vee.” This step addressed the problem decomposition, morphological analysi, and morphological analysis revisited research questions, and validated the corresponding hypotheses.

Then, cross-consistency assessment methods have been used to document relational data between alternatives identified in the problem decomposition, thus establishing the combinatorial logic that drives the problem synthesis into a number of internally consistent operational configurations. The traditional binary scale used to study the compatibility between alternatives in the original MA formulation has been modified to describe the relative consistencies at each level of decomposition identified in the previous step. Cross-consistency assessments based on probabilistic or likelihood representations have been introduced to examine the nature of the internal relationships between elements of the problem and to reduce the number of scenarios to an operationally relevant set. Such cross-consistency assessment schemes also enabled encoding relational data with higher resolution scales to capture more complex interactions. This step addressed the cross-consistency assessments and cross-consistency assessments revisited research questions, and validated the corresponding hypotheses.

With this, the first leg of the “Vee” diagram is complete and the parametric representations research question is fully addressed as depicted in Figure 21.

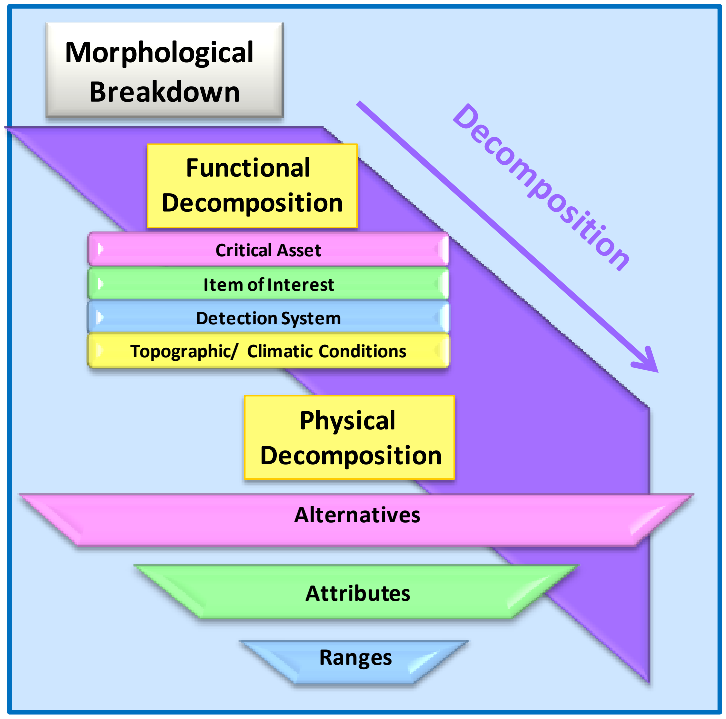


Figure 21: Morphological Breakdown of the Problem

4.4. <u>How to Define a Relevant Operational Scenario</u>

The definition of a parameter space in the SLMM allows the analyst to create parametric representations for each of the element alternatives prescribed in the HLMM. All alternatives for a given element are represented by a common vector of parameters, and are distinguished from each other by means of the specific values selected. This provides the analyst with the flexibility to create an operational scenario to be modeled in two ways. One way is to use the HLMM, and to directly select prescribed alternatives for which parametric representations have been provided. The other way is to use the SLMM and to dial in one by one the values of the attributes for all elements, hence departing from prescribed alternatives and formulating notional ones. A combination of the above two ways for defining a candidate scenario may also be used, for instance by selecting alternatives from the HLMM as a starting point, and then implementing small perturbations by altering parameter values. The selected alternatives and parameter values for the candidate scenario can finally be assessed for compatibility using the Cross-Consistency Matrices to determine whether the considered scenario is valid. If it is not, alternatives and/or parameter values may be changed until an adequate combination is obtained.

In other words, the information contained in the various morphological matrices can be used to define scenarios for surveillance and protection missions in the context of homeland security, as follows. The alternatives identified in the HLLM are modeled using intrinsic design relationships and physics-based representations involving the various attributes and their ranges of values or settings defined in the corresponding SLMM for each element considered. The information concerning the compatibility of alternatives depending on the values or settings of their attributes, contained in the HLCCM and the SLCCMs, is then used to eliminate those combinations of alternatives, attributes, and values or settings that are contradictory with each other or that do not represent realistic situations. Only those scenarios presenting a meaningful operational context remain. The said scenarios may then be modeled and simulated in a modeling and simulation (M&S) framework so as to derive useful information regarding customized detection architecture solutions able to protect a critical area of interest in a specific geographic and climatic environment.

CHAPTER V

IMPLEMENTATION – MODELING OF A PROOF-OF-CONCEPT SCENARIO

Once the Morphological Matrices and the Cross-Consistency Matrices are completely finalized, the information they contain may be used to develop a modeling and simulation environment, or, more precisely a multidisciplinary design and optimization framework, where any number of notional "what-if" scenarios can be played.

This section addresses the modeling, simulation, and optimization, solutions analysis, and what-if analysis research questions, along with the corresponding optimization method, optimization parameter settings, and solutions benchmarking and accuracy research questions, and serves as a test to the corresponding hypotheses.

In this section and in the remainder of this work, we will consider the example of a Customs and Border Protection (CBP) mission which is well suited for the design, modeling, simulation, and optimization of distributed systems detection architectures. Indeed, the surveillance of large terrains and geographic borders using a limited detection capability has always been an issue in the defense and protection community of many developed countries. Geographic borders are usually very long and may go through isolated areas with little or no inhabitants. As a result, it is almost always impossible to completely secure each and every mile of border against potential smugglers, drug dealers, illegal border crossings, and malevolent entities who are trying to enter a country to exploit or harm its people or infrastructures. In addition, the costs associated with the development of efficient architectures of protection systems may be so large that they become prohibitive. A compromise must therefore be made and a balance must be found between the level of acceptable protection and the cost governments are willing to invest. However, different countries may have different preferences when it comes to compromising between protection efficiency and cost, mostly depending on their histories and related experiences. Capturing such disparities is not straightforward and often requires the use of a parametric analysis or cost/benefit analysis. In addition, different preferences result in different protective solutions. Therefore, there exists a need for a method able to provide customized protection alternatives for varying levels of performance and cost. Such a method is found in the domain of optimization which enables the design of protection architectures that perform efficiently in a wide range of external conditions. These architectures are defined as Pareto optimal and are obtained by performing parametric optimizations.

In the context of CBP missions, optimizing sensor location over large terrains is a crucial need for the defense and protection community. In this context, coverage and cost are fundamental and widely accepted metrics to evaluate the performance of surveillance systems in domains such as intrusion detection and border surveillance. The major goal in such applications is to detect intruders as they penetrate a protected region or before they cross a border. This constitutes a barrier coverage problem. Unlike other applications, the aforementioned problems do not require the coverage of each and every point of a given region but rather the coverage of that portion of the region through which intruders can penetrate. As such, the sensors can be deployed as a barrier to decrease the cost and achieve an acceptable coverage.

The present research concentrates on the design and optimization of multispectral multi-platform sensor architectures for border surveillance missions. More precisely, it considers the example of illegal crossings of the Arizona- Mexico border. The current study formulates an optimization problem that is then investigated using both a modified Genetic Algorithm and a modified Particle Swarm Optimization (PSO) algorithm. Appropriate sets of parameters for both optimization algorithms are derived and their ability to reliably find adequate detection architectures that provide the maximum detection coverage at the minimum cost is compared. Finally, a parametric analysis is performed on the objective function so as to capture the unknown customer preference between performance and cost of the detection architecture, and to determine a set of Pareto optimal or non-dominated solutions capable of satisfying constraints on both performance and cost.

Figure 22 shows a schematic representation of the modeling, simulation, and optimization environment developed to analyze the CBP mission scenario.

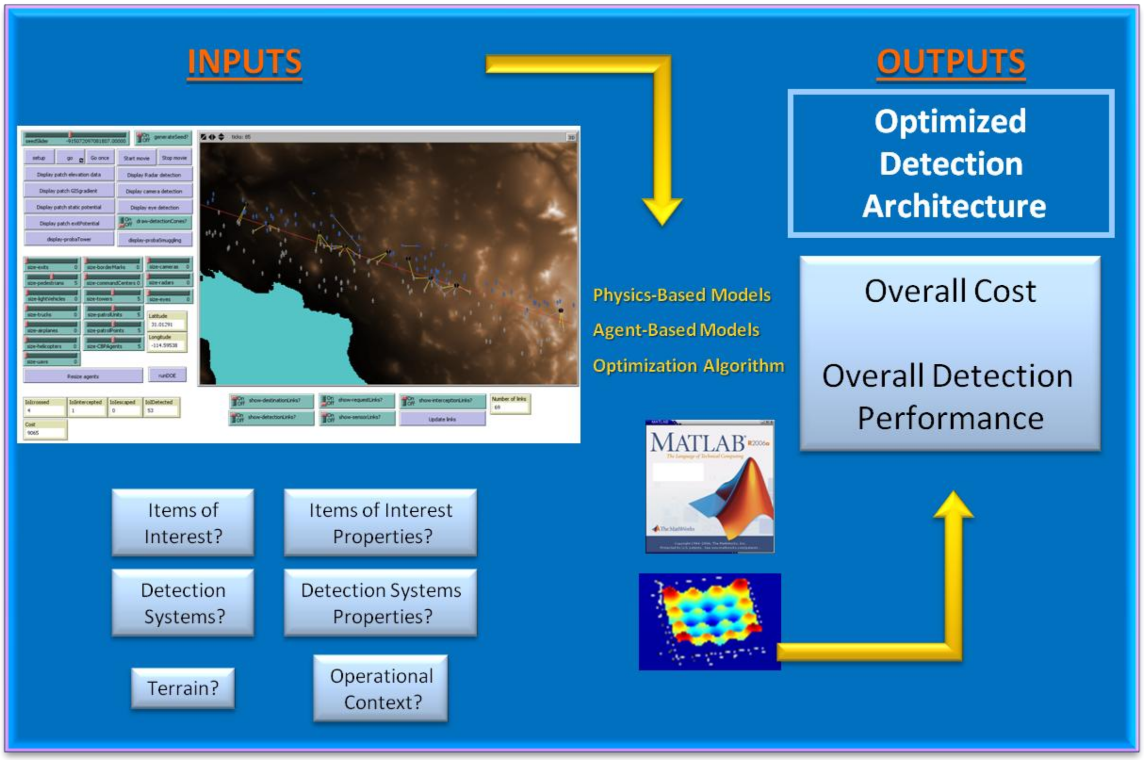


Figure 22: Schematic Representation of the Modeling, Simulation, and Optimization Environment

The picture on the left of Figure 22 represents a map of the terrain at the Arizona-Mexico border for the CBP mission scenario considered. The goal of the M&S and optimization environment is to determine the optimum combination of detection systems in terms of types, numbers, properties, and positions on the theater of operations that yields the maximum detection performance at the minimum cost. For this, the available information concerns the types and categories of detection systems, pre-selected positions on the terrain where those detection systems will be preferentially located when optimizing the detection architecture, and the various items of interest in the surveillance mission under study. Other necessary inputs are the geographical properties of the theater of operations, as well as the operational context in which the scenario is taking place, which includes the time required for potential interception of detected suspicious items of interest. Finally, as can be seen on the right of Figure 22, MATLAB is used to develop an optimization algorithm able to reliably find detection architectures providing adequate coverage at reasonable cost in the context of the CBP mission scenario of interest. The subsequent sections will detail the selection of the modeling, simulation and optimization framework elements for the analysis of the CBP mission scenario, and will go through each step of the methodology leading to the reliable design and optimization of appropriate detection architectures.

5.1. Selection of the Modeling, Simulation, and Optimization Framework

In order to perform the above analysis, it is necessary to select both an appropriate modeling and simulation framework and an optimization environment where the problem can be accurately represented and optimized. Given the emergent behavior of the detection system-of-systems under study, an agent-based modeling and simulation (ABM&S) framework is required. Then, in order to perform the optimization of the detection architecture, an optimization framework is selected. The selection of both the ABM&S and the optimization frameworks is performed by examining candidate platforms and tools, and rating them according to a set of selection criteria. The programming platforms and tools considered are Eclipse, MATLAB, NetLogo, and other ABM tools (such as SWARM, RePast, and MASON). The selection criteria against which the above programming platforms and tools are compared are availability, learning curve, ease of use, computational expense, fidelity of physics and agent behavior, availability of a visualization capability, level of user interactivity, and compatibility with other software used for this research (especially Excel for the Morphological Analysis and for post-processing). NetLogo and MATLAB turn out to be the most suitable tools for modeling, simulation, and optimization (MS&O) of the CBP problem of interest, essentially because they enable the user to build up models and optimization algorithms from scratch and they provide a high flexibility and transparency in the programming of the modeling, the simulation, and the optimization routines.

5.1.1. Definition of Selection Criteria

Firstly, due to cost and implementation constraints, the MS&O tool need to be available, either publicly or widely used in the industry.

Secondly, due to time constraints, it is required that the learning curve of the tools be reasonable, which translates into MS&O tools familiar to the author.

Thirdly, the elements of the scenario (especially the items of interest and the detection systems), described in previous sections, need to be modeled using physics-based models. In other words, models are supposed to be built with mathematical equations or basic concepts, relating to the design, the performance, the behavior, etc, of the parameters of the study.

Fourthly, the M&S tool is required to be agent-based so as to be able to simulate the behavior of the items of interest and of the detection systems over time, while the optimization framework needs to be sufficiently flexible and transparent to develop custom optimization algorithms.

Fifthly, the final product is intended to provide a visual support for in-house demonstration purposes, and to be interactive. Therefore, other desired features of the M&S tool are its visualization capability, its user friendliness, and its compatibility with other M&S software that may be already in use at governmental industries.

Sixthly, the MS&O tools need to be compatible with Excel and/or other mathematical software. Indeed, such tools may come in handy for calculation purposes and/or for recording the results at each step of the methodology in case they are needed later in the process.

Lastly and maybe most importantly, due to the modeling and simulation requirements associated with the problem under study, it can only be a good idea to consider the worst case scenario and assume that the MS&O software will have to deal with computationally expensive simulations and optimizations. Therefore, the last criterion for selection is that the MS&O tools have acceptable memory capacities and a sufficient computational power.

To summarize, the MS&O tools are required to have the following characteristics:

- Readily available
- Reasonable learning curve
- Familiar to the author
- Physics-based / Agent-based
- Flexible and Transparent
- Visualization Capability
- Interactive Capability
- Compatibility with Excel and mathematical Software
- Computationally friendly

5.1.2. Description of Candidate ABM Programming Platforms and Tools

Several relevant ABM and optimization programming platforms and tools potentially satisfying one or more of the aforementioned criteria are identified through literature search. These are:

- SWARM
- RePast (Recursive Porous Agent Simulation Toolkit)
- MASON
- NetLogo
- Eclipse
- MATLAB

Each of them is briefly described in Appendix H.

5.1.3. Comparison and Selection of the MS&O Framework

Based on the characteristics of the relevant ABM&S and optimization programming platforms and tools described in previous sections, a table can be created to compare the ability of these tools to satisfy the selection criteria identified previously. The results are depicted in Table 7.

Table 7: Comparison of Relevant ABM&S and Optimization Programming Platforms and Tools According to Selection Criteria

Tools *Criteria*	Eclipse	MATLAB	NetLogo	SWARM	Repast	MASON
Available	✓	✓	✓	✓	✓	✓
Reasonable learning curve	✓	✓	✓	✓	✓	✓
Easy to use	✓	✓	✓	✓	✓	✓
Computationally friendly	✓	✓	✓	✓	?	✓
Physics-based / Agent-based	✓	✓	✓	✓	✓	✓
Flexible / Transparent	✓	✓	✓	✓	✓	✓
Visualization Capability	✓	✓	✓	✓	✓	✓
Interactive Capability	✓	✓	✓	?	✓	✓
Compatibility with Excel and Mathematical Software	?	✓	✓	✓	✓	?

Legend:

- ✓ does have that property
- ? not specified

As can be noticed from Table 7, **MATLAB** turns out to be the most appropriate optimization tool for the problem at hand, while **NetLogo** turns out to be the most appropriate ABM&S tool to model and simulate detection architectures under various operational conditions. Therefore, NetLogo and MATLAB are used to model, simulate and optimize the detection architecture for a proof-of-concept scenario of interest. This test operational scenario is constructed from the Morphological Matrices, and is simulated using the compatibility information contained in the Cross-Consistency Matrices. A description of the proof-of-concept scenario and of the models of the terrain, the defended asset, the items of interest, and the detection systems is provided in subsequent sections.

5.2. Morphological Decomposition of the Customs and Border Protection Mission Scenario

5.2.1. Problem Decomposition

Similarly to the CAP mission analyzed in previous sections, the decomposition process of the CBP mission begins at the highest level where the problem construct is decomposed into its primary components, namely the border, the detection systems, and the items of interest trying to cross the border. Additionally, the operational environment is recognized as a key driver impacting performance at all levels. It is represented as an additional element of the system, both in terms of topographic and climatic conditions. Each of the above elements is then further decomposed into sub-elements, effectively implementing a new level in the hierarchical structure of the problem construct. Finally, for each sub-element, a list of possible alternatives is identified and documented in the High Level Morphological Matrix (HLMM). This is shown in Figure 23 for the Customs and Border Protection (CBP) mission.

ELEMENT	SUB-ELEMENT	ALTERNATIVES				
CRITICAL ASSET	*Type*	Full Land Border	Border Sector	Border Control Point		
TOPOGRAPHIC CONDITIONS	*Mountain*	Isolated Mountain	Chain of Mountains			
	Body of Water	River	Lake	Sea/Ocean/ Coastline		
	Forest	Very small	Small	Medium	Large	Very large
	Urban Environment	Cluster of houses	Town	City	City + Neighborhoods	Metropolis
CLIMATIC CONDITIONS	*Atmosphere*	Tropical	Mid-latitude Summer	Mid-latitude Winter	Subarctic Summer	Subarctic Winter
	Aerosol State	Clear	Hazy			
ITEM OF INTEREST	*Air*	General Aviation Aircraft	Crop Duster	Ultralight	Motorized Glider	Unmanned Air Vehicle
	Ground	Pedestrian	Group of Pedestrians	Car	Van	Truck
		Motorcycle	Bicycle	Horse		
SENSOR	*Active Radar Air/Ground*	Radar A	Radar B	Radar C	Radar D	…
	Camera Air/Ground	Camera A	Camera B	Camera C	Camera D	…

Figure 23: High Level Morphological Matrix for the Customs and Border Protection Mission

The next part of the decomposition deals with the definition of sets of attributes and the associated ranges for their values with which *parametric representations* of system elements can effectively be made with a modeling and simulation capability in mind. The selection of attributes and ranges is such that it adequately captures the whole spectrum of alternatives prescribed for each "Element" in the HLMM, and also allows for additional parametric definitions of other alternatives not yet considered. These attributes may be continuous or discrete, and therefore ranges of values or other adequate representations of variables must be defined accordingly. The definition of the attributes and their representations for each sub-element of the problem is documented in Sub Level Morphological Matrices (SLMMs).

In the CBP mission, a salient feature for the parametric representation of the border is its relative importance to the decision maker so that an appropriate level of required protection can be specified for any specific case. Consequently, the problem is best approached by giving the decision maker the opportunity to define a level of protection to be achieved with the detection SoS. In this context, the focus being only on the detection of potential illegal border entries, a relevant attribute is the threshold detection probability of an item of interest within a given distance from the border. Moreover, the level of protection might be a function of the performance of protection systems that may already be in place along the border. In order to capture the efficiency of said known or unknown systems, field experts consider the time needed to deal with the detection or interception of items of interest using those systems. This yields a distance before which a tentative illegal entity has to be detected and identified to be efficiently handled. This distance before which items of interest must be detected may be represented as a band of protection along the border. The band of protection is the sum of the "staging zone", located south of the border, and of the "crossing zone", located north of the border as depicted in Figure 24. Three other zones are considered around the border, namely the "sanctuary zone" south of the border, the "hide zone" north of the border, where the items of interest tend to hide so as not to be intercepted by local authorities, and the "transit zone" north of the border, where the items of interest are free to move in the United States and are assumed to blend in the local population.

Figure 24: Definition of the Arizona-Sonora Border

Table 8 provides the attributes and ranges in the SLMM for the "critical asset" in the CBP mission.

Table 8: Attributes and Ranges for the Critical Asset in the CBP Mission

	ATTRIBUTE	UNIT	MIN VALUE	MAX VALUE
CRITICAL ASSET	*Distance from Border of Sanctuary Zone*	*km*	10	20
	Distance from Border of Staging Zone	*km*	1	10
	Distance from Border of Crossing Zone	*km*	1	15
	Distance from Border of Hide Zone	*km*	15	30
	Distance from Border of Transit Zone	*km*	> 30	

The definition of attributes for the "item of interest (IoI)" element necessitates taking into account requirements for the modeling of detection systems, especially the calculation of their performance metrics depending on the nature of the "item of interest". Therefore, the wide variety of potential items of interest can be captured into three attributes for the CBP mission. Table 9 provides attributes and ranges in the SLMM for the "item of interest".

Table 9: Attributes and Ranges for the "Item of Interest" in the CBP Mission

	ATTRIBUTE	UNIT	*MIN VALUE*	*MAX VALUE*
GROUND ITEM OF INTEREST	*Average Speed*	*km/h*	5	130
	Average Radar Cross Section Pedestrians	m^2	0.5	4
	Average Radar Cross Section Vehicles	m^2	2	100

A similar approach can be adopted to define appropriate attributes for the terrain and the climatic conditions. The attributes and ranges in the SLMMs for the terrain and for the climatic conditions for the CBP mission are the same as those for the CAP mission and are summarized in Table 3 and Table 4 respectively.

The attributes characterizing the detection systems of interest, namely active radars and cameras, fall out directly from intrinsic design parameters of those systems. For instance, Table 10 and Table 11 provide the attributes and ranges in the SLMMs for the ground radars and for the optronic systems considered in the CBP mission.

Table 10: Attributes and Ranges for the Ground Active Radar in the CBP Mission

RADAR OPTION	*Type*	Radar A			
		Radar B			
		Radar C			
	Nature	Mobile			
		Fixed			
	Rotating?	Yes			
		No			
	ATTRIBUTE	**UNIT**	**MIN VALUE**	**MAX VALUE**	
	Frequency	MHz	X-band		
	Losses + Noise Factor	*dB*	3	21	
	Bandwidth	*MHz*	1	300	
	Elevation Tilt Up	°	0	20	
	Elevation Tilt Down	°	0	20	
	Elevation Coverage	°	-20	140	
	Azimuth Beamwidth	°	4	4.3	
	Field of View	°	100	360	
	Minimum Detectable Target Speed	*km/h*	0	50	
	Maximum Number of Targets Tracked	-	60	100	
	Range for pedestrians detection	*km*	22	31.5	Radar A
			16	26	Radar B
			8	15	Radar C
	Range for vehicles detection	*km*	22	36	Radar A
			18	30	Radar B
			11	17	Radar C
	Range for aircraft detection	*km*	9	36	Radar A
			7	30	Radar B
			7.4	17	Radar C
	Scanning Rate If Applicable	°…*s*	32	64	
	Cost	*Kdollars*	150	850	

Table 11: Attributes and Ranges for the Optronic Systems in the CBP Mission

OPTRONIC SYSTEM	*Type*	1		
		2		
		3		
	Bi-field	Yes		
		No		
	ATTRIBUTE	**UNIT**	**MIN VALUE**	**MAX VALUE**
	Wavelength Infrared 1	*µm*	10	10
	Wavelength Infrared 2	*µm*	3	5
	Wavelength Infrared 3	*µm*	8	12
	ARRAY SIZE			
	Number of Horizontal Pixels	-	480	480
	Number of Vertical Pixels	-	640	640
	OPTRONIC SYSTEM			
	Focal Dimension	*cm*	1	100
	Optical Diameter	*cm*	3	15
	Detection Range	*km*	1	20
	Sensibility (NedT)	*mK*	1	150
	Integration Time Infrared 1	*ms*	40	40
	Integration Time Infrared 2	*ms*	20	20
	Integration Time Infrared 3	*ms*	5	5

As mentioned earlier, this section deals with the problem of designing and optimizing multispectral 3D detection architectures for border protection and intrusion detection at the Arizona-Sonora border. However, sensor placement requires accurate yet computationally efficient sensor detection models, as well as a sufficiently fine representation of the terrain topography.

5.2.2. Cross-Consistency Assessments

In order to define relevant scenarios for the CBP mission, the morphological analysis of the problem is complemented by probabilistic cross-consistency assessments of the various elements identified. Similarly to the CAP mission analysis, these cross-consistency assessments are based on the likelihood scale defined in section 4.2.1.1. This likelihood scale is intended to be flexible enough for decision makers to analyze a wide range of homeland security missions. Using empirical data and specific knowledge about the situation they are trying to model and simulate, experts in the field of homeland security can determine probabilistic assessments between the various elements involved in the CBP mission. From these cross-consistency assessments, they may then define relevant scenarios so as to gain insight into the structure and the performance of the distributed detection system architecture required to perform the CBP mission of interest.

For instance, past experience and recent knowledge about the operational situation at the Arizona-Sonora border may lead experts to believe that drug dealers riding a camel in the desert are highly unlikely. Therefore, it is unnecessary to model this situation in an operational scenario. Similarly, it may be very unlikely that pedestrians will hide in forested areas in the Arizona-Sonora border region due to the lack of forests. However, although it may be unlikely to see groups of pedestrians climbing mountains in an operational scenario, it is rather likely than they will follow river streams, and very likely that they will walk along successful smuggling trails. Finally, in a CBP mission scenario, it is always true that pedestrians will avoid border control check points as they try to illegally penetrate the United States of America.

5.2.3. Scenario Definition – Example

The information contained in the morphological matrices for the CBP mission, depicted in Figure 23, and Table 8 to Table 11, may be combined with the likelihood cross-consistency assessments developed in the previous section to define relevant operational scenarios for the CBP mission. In practice, the information provided by the problem decomposition and the compatibility assessments may be used to analyze a detection architecture in the modeling and simulation environment described in the following section. This implies determining the operational efficiency of a newly designed detection architecture, analyzing the sensitivity of its performance to changes in the operational situation, studying the impact of changes in its structure and composition on its detection capability, or complementing it with carefully selected fixed or mobile detection systems to enhance its operational effectiveness when required.

A sample scenario, taking place along the Southwest border between the state of Arizona in the United States of America and the state of Sonora in Mexico, in the Douglas sector area of operations, is described subsequently.

The Douglas Station is located within the Southwest Cochise County and covers approximately 1400 square miles. The station's area of operations includes approximately 47 linear miles of the Arizona-Sonora border. There are currently 515 Border Patrol agents assigned to the Douglas Station. The communities of Douglas, Pirtleville, Elfrida, and McNeal, AZ are within the station's area of operations. The City of Douglas shares the border with Aqua Prieta, MX. The terrain of the area is relatively high desert, with numerous washes, and is bordered by the Dragoon and Mule Mountains to the west, and the Chiricahua, Pedregosa, and Perillla, and Peloncio Mountains to the east. This is depicted in Figure 25.

Figure 25: Douglas Sector

Consider Ali Mohammed, a person of interest, born in Saudi Arabia, and trained in Afghanistan and Pakistan. The group by which he was recruited decided to send Ali to the United States via Mexico. The group contacts a smuggler in Mexico City, who has contacts with other smuggling groups throughout northern Mexico. The group also decides that the best way for Ali to enter the U. S. is illegally, thereby the U.S. having no record of Ali's entry into the United States.

Since Ali is very intelligent, having graduated from some of the better schools in Saudi Arabia and Europe, it is thought that having him being fluent in Spanish would be a great asset to Ali, both in Mexico and in the United States, especially if apprehended by the Border Patrol. Since Ali was born of a Saudi Father and Hispanic Mother, he could possibly pass himself off as a Mexican National and be repatriated to Mexico. If he were repatriated to Mexico, rather than waiting for a deportation hearing as a Saudi National, he could re-enter the United States quickly. Thus, if Ali persists in claiming that he is a Mexican National, with his looks and Spanish fluency, the Border Patrol will grant him a “Voluntary Return” to Mexico. The group which recruited Ali is supposed to be knowledgeable about the Border Patrol assets and intelligence resources.

Ali is furnished an altered Mexican Passport, is flown from Mexico City to Hermosillo and then takes a bus to Agua Prieta. The group does not want to fly him directly into the area, because of the Mexican Policia de Fronteriza (Border Police). As Ali arrives in Agua Prieta, he is met by a person hired by the smuggling group, unknown to the Mexican Policia de Fronteriza. He is taken to a “safe house” where he is temporarily housed until the smuggling group is able to gather enough people for a trip north into the United States. At this point, Ali begins to speak nothing but Spanish.

It is planned that after 3 days in the “safe house”, the group will be taken west out of Agua Prieta, approximately 30 miles and dropped off in a mountainous area. The group will then be met by a Guide who is to take them to a location on Interstate 19, where they are to be picked up by a driver in a blue van and taken to Tucson, Arizona. Before that, the group will have to walk through the mountains for about 5 hours, setting off several Border Patrol sensors, which had been placed along the trail.

The goal of the simulation of the previous scenario in a modeling and simulation environment may be twofold:

- Testing the operational efficiency of a newly designed detection architecture to determine whether it responds to a specific need: in the scenario above, this means detecting Ali and its group before they arrive in Tucson.
- Complementing an existing detection architecture with carefully selected fixed or mobile detection systems to enhance its ability: in the scenario above, this means being able to detect Ali and its group as they are dropped off in the mountainous area, or as they walk through the mountains along well known smuggling trails, or as they are taken to Interstate 19 by the Guide, or as they are picked up by the blue van to be driven to Tucson.

5.3. Modeling of the Customs and Border Protection Mission Scenario

5.3.1. Modeling and Simulation Environment

5.3.1.1. Setup and Main Dimensions

The terrain is modeled using Shuttle Radar Topography Mission (SRTM) data obtained from Google Earth [461]. It is imported in the agent-based programmable modeling environment NetLogo through a "map creator" specifically developed for this application. This is depicted in Figure 26, which shows the actual terrain as depicted in Google Maps and in the NetLogo environment where the border is modeled as an angled red line separating the American territory (located north of the border) from the Mexican territory (located south of the border).

Figure 26: Terrain Topography and Border Modeling in the CBP Mission

The above environment is called the "world". The "world" is a two-dimensional environment made out of "patches", similar to a pixel the size of which is representative of a physical distance in the real world. In the NetLogo environment developed, each patch corresponds to a square of size 1 km x 1 km. The "map creator" automatically re-samples the SRTM elevation dataset by converting high-resolution meshes to lower fidelity meshes adapted to the 1-km length scale of the modeling and simulation environment. Each “patch” in the “world” is thus characterized by three parameters: a latitudinal location and a longitudinal location which define the two-dimensional position of the patch on the terrain, and an elevation value which defines a third vertical dimension. The elevation value of each terrain patch is obtained by averaging the elevation values of the nearest data points provided by the SRTM data. The average elevation data associated with each patch yields an elevation map from which a gradient map of the terrain elevation may be derived. The gradient map of the terrain elevation is shown in Figure 27.

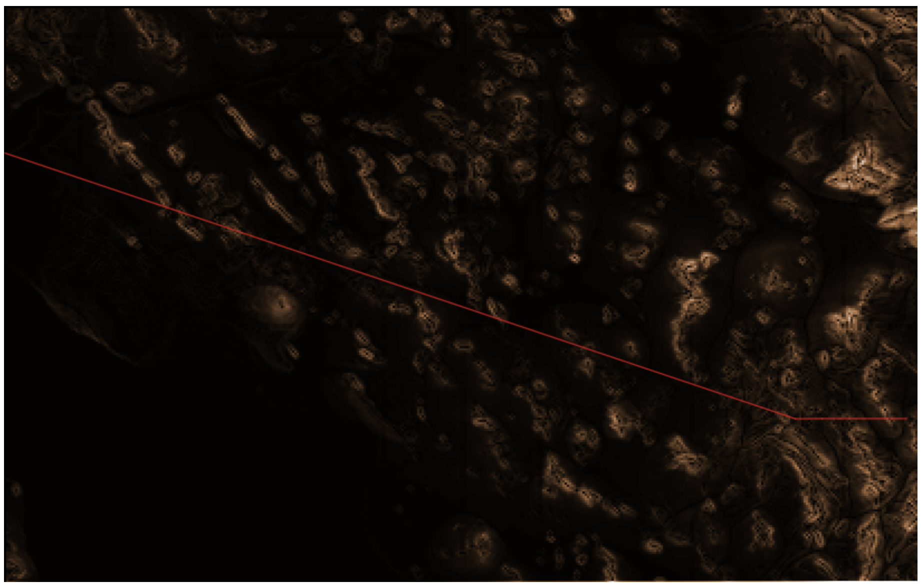


Figure 27: Gradient Map of the Terrain

The gradient map encompasses east-west and north-south data that may then be used to determine the gradient-dependant speed of motion of the “agents” in the “world”. This will be described in the following sections. With the aforementioned conventions, the horizontal extent of the “world” corresponds to 408 km while the vertical extent of the world is 258 km. In terms of longitude and latitude, this translates into an area extending from 110.5° West to 114.8° West, and from 30.6° North to 33° North. Moreover, 1 km in the real world corresponds to 0.1° latitude in the vertical dimension. This is depicted in Figure 28.

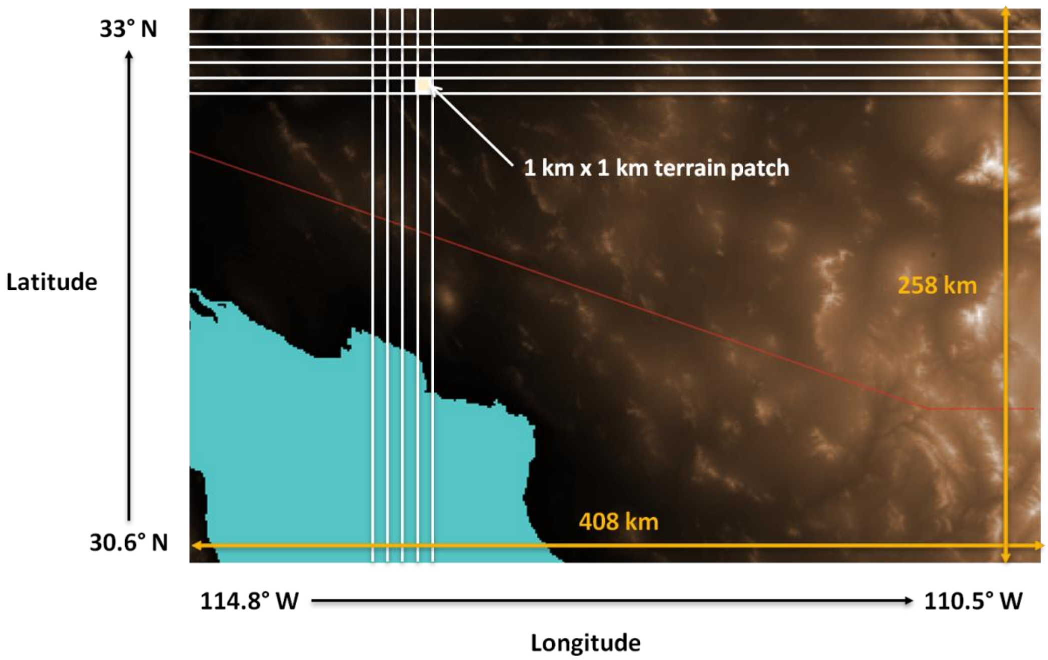


Figure 28: NetLogo Environment and Main Dimensions

5.3.1.2. Border Definition and Band of Detection

In the NetLogo modeling and simulation environment, the border is modeled by Equation 1:

$$\text{Lat.} = \begin{cases} -0.28157895 * \text{Long.} + 0.074736842 & \text{for Long.} \leq -111^{\circ} \text{ (Western portion of the border)} \\ 31.33^{\circ} & \text{for Long.} \geq -111^{\circ} \text{ (Eastern portion of the border)} \end{cases}$$

Equation 1

In order to be able to detect illegal border crossings, a band of detection is defined across the border line. The detection band encompasses the "crossing zone" and the "staging zone" depicted in Figure 24, and extends 15 km on the American side and 10 km on the Mexican side. This corresponds to a longitudinal location varying from 110.5° West to 114.8° West, and to a latitudinal location varying by either 0.15° from its value at the border along the inclined portion of the border (longitude smaller than 111° West) or

0.1° from its value at the border along the horizontal portion of the border (longitude larger than 111° West), i.e. between 31.23° and 31.48° in the last case. This corresponds to a 0.15° and 0.1° variation in latitude on the American side and on the Mexican side respectively.

The problem is now to determine the combinations of sensor systems in terms of numbers, types and locations (latitudes and longitudes), that maximize the surface coverage of the band of detection while minimizing the cost of the resulting protection architectures. While the band of detection extends across the border on the Mexican side, the sensor systems have to be placed only on the American side of the band of detection. Therefore, their longitudinal location still ranges from 110.5° West to 114.8° West but their latitudinal location now varies by 0.15° from its value at the border along the inclined portion of the border, and from 31.33° to 31.48° along the horizontal portion of the border. However, a potential position for the detection systems on the American side of the band of detection is represented by a terrain patch in the NetLogo environment, and a terrain patch models an area of 1 km^2. Therefore, there exit about 10,700 possible positions for the detection systems inside the band of detection. This is depicted in red in Figure 29, along with the band of detection represented in blue and a white terrain patch of 1 km^2 modeling a potential position for the detection systems inside the band of detection. The part of the band of detection in which the detection systems may be positioned corresponds to the crossing zone defined in Figure 24.

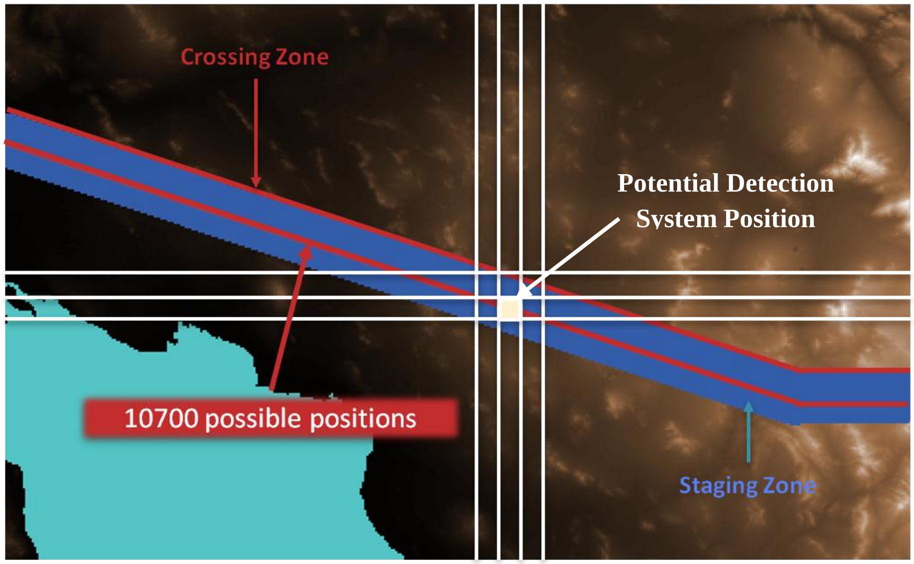


Figure 29: Band of Detection

5.3.1.3. Coverage Efficiency and Architecture Coverage

In accordance to the aforementioned positioning constraints, it is worthwhile defining the notion of coverage as it is used in this research for both a single detection system and a detection architecture of distributed detection systems.

5.3.1.3.1. Coverage Efficiency

Consider first a detection system positioned at the center of a patch in the NetLogo environment. This patch corresponds to a position inside the crossing zone and is associated with an elevation value E_d. In order to determine the coverage efficiency of a detection system, it is necessary to create a "line-of-sight" algorithm which determines the terrain patches that are visible from the position of the detection system in the theater of operations. Consider a detection system located at a position x on the theater of operations. This detection system is characterized by its design detection range R and its design elevation angle θ. These two attributes define the half-sphere of detection of the sensor. The "line-of-sight" algorithm first determines the set S of terrain patches located within the base of the half-sphere of detection of the sensor by correlating the two-dimensional position of the terrain patches with the extent of the base of the half-sphere of detection. Consider a terrain patch *i* located within the half-sphere of detection of the sensor, characterized by the elevation value E_i. Define the straight line between the sensor and the terrain patch *i* as the "line-of-sight" and projects it on the two-dimensional terrain. Then, consider the set *P* of all other terrain patches located within the half-sphere of detection of the sensor and along the line-of-sight between the terrain patch *i* and the detection system. Each terrain patch in the set *P* is characterized by an elevation value E_p. For each terrain patch in the set *P*, the "line-of-sight" algorithm determines whether the elevation value E_p associated with this patch is smaller or larger than the elevation value E_i associated with the terrain patch *i* of interest. If all the elevation values E_p are smaller than the elevation value E_i, then the detection system actually sees patch *i*. In this case, patch *i* is in the "line-of-sight" of the detection system as notionally shown in Figure 30. In Figure 30, the blue square is the detection system located on the side of a mountain, the blue half-sphere is the half-sphere of detection of the sensor, the red circle is the projection of the half-sphere on the two-dimensional terrain (or base of the half-sphere); the green square is the terrain patch *i* of interest located on a flat area; the purple squares are terrain patches in the set *P*; the white vertical arrows starting from the detection system, the terrain patch *i*, and the terrain patches in set *P* are modeling the elevation values of the corresponding patches; the orange line is the line-of-sight between the notional height of the detection system given by the corresponding patch elevation E_d and the notional eight of the terrain patch *i* given by E_i, and the dashed orange line is the projected line-of-sight on the two-dimensional terrain (or on the base of the half-sphere of detection of the sensor).

On the contrary, if one of the elevation values E_p is larger than the elevation value E_i, then the detection system cannot see patch *i*. In this case, patch *i* is "out-of-sight" from the detection system position, as notionally represented in Figure 31. In Figure 31, the same symbol and color conventions as in Figure 30 are used. The aforementioned process is repeated for each terrain patch *i* in the set *S* of terrain patches within the design detection range of the sensor.

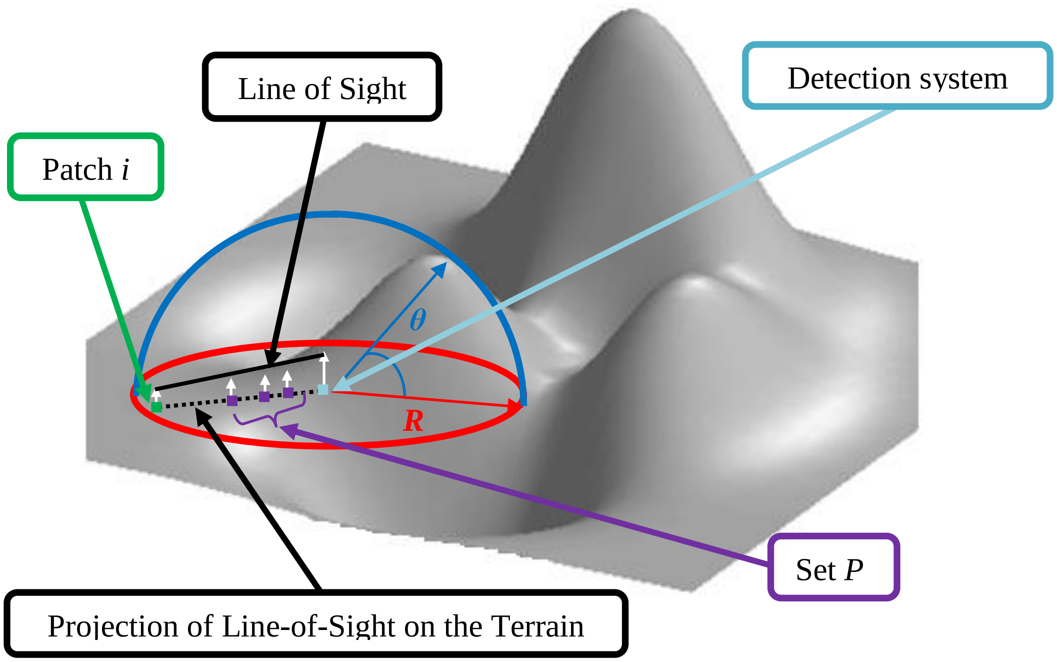


Figure 30: "Line-of-Sight" Algorithm: "Line-of-Sight" Case

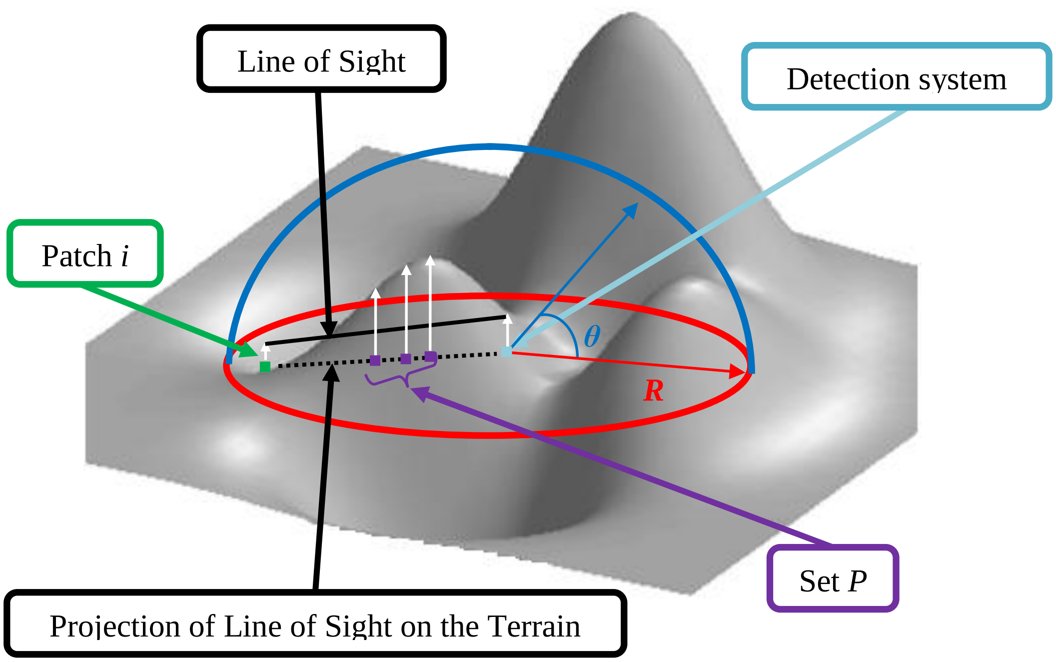


Figure 31: "Line-of-Sight" Algorithm: "Out-of-Sight" Case

Using the "line-of-sight" algorithm, it is possible to determine three different regions of patches inside the half-sphere of detection of a detection system of interest. Consider the notional situation depicted in Figure 32. In Figure 32, the blue area represents the terrain patches inside the band of detection. Superimposed on the blue area are a yellow area and a green area. The yellow area corresponds to the terrain patches within the band of detection that are "out-of-sight" or not visible from the position of the detection system. In other words, the yellow patches are not actually seen by the detection system due to topographic obstacles in its line of sight. Finally, the green area represents the terrain patches within the band of detection that are in the "line-of-sight" or visible from the position of the detection system. In other words, the green patches are actually seen by the detection system because there are no topographic obstacles in its line of sight. The coverage efficiency of the detection system may then be defined as the ratio of the green area to the green + yellow areas as shown in Equation 2.

$$\text{Coverage Efficiency} = \frac{\text{Green Patches}}{\text{Green+Yellow Patches}}$$

Equation 2

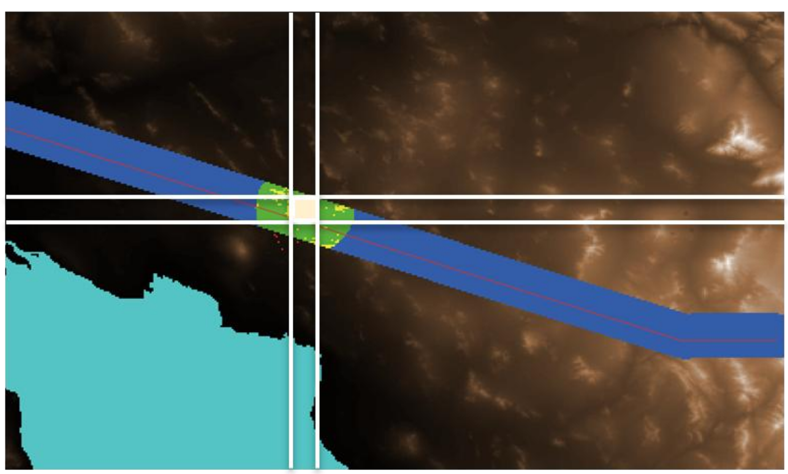


Figure 32: Coverage Efficiency of a Single Detection System

5.3.1.3.2. Architecture Coverage

Consider the notional detection architecture of distributed detection systems depicted in Figure 33 in the NetLogo environment. In Figure 33, the white circles with the grey symbols inside are the detection systems. Using the “line-of-sight” algorithm for each sensor in the detection architecture, it is possible to determine three different regions of patches. The blue area represents the terrain patches inside the band of detection. Superimposed on the blue area are a yellow area and a green area. The yellow area corresponds to the terrain patches within the band of detection that are not visible from any detection system in the detection architecture due to topographic obstacles in their lines of sight. The green area represents the terrain patches within the band of detection that are visible from one or more detection systems in the detection architecture. In order to determine the actual coverage of the detection architecture, it is necessary to account for any overlap in coverage provided by the detection systems composing the architecture. To do so, the “line-of-sight” algorithm is coupled to an “overlap” algorithm which ensures that terrain patches visible from more than one detection system in the architecture are counted as being seen by the whole architecture only once. Similarly, the “overlap” algorithm ensures that terrain patches that are not visible from any detection system in the architecture are counted as being not seen by the whole architecture only once. This avoids counting multiple times “visible” and “not visible” patches (i.e. green and yellow patches respectively), which would bias the resulting architecture coverage value. Indeed, counting too many “visible” patches would inappropriately increase the architecture coverage value, while counting too many “not visible” patches would inappropriately decrease the architecture coverage value. The coverage of the detection architecture may then be defined as the ratio of the green area to the green + yellow + blue areas as shown in Equation 3.

$$\text{Architecture Coverage} = \frac{\text{Green Patches}}{\text{Green+Yellow + Blue Patches}}$$

Equation 3

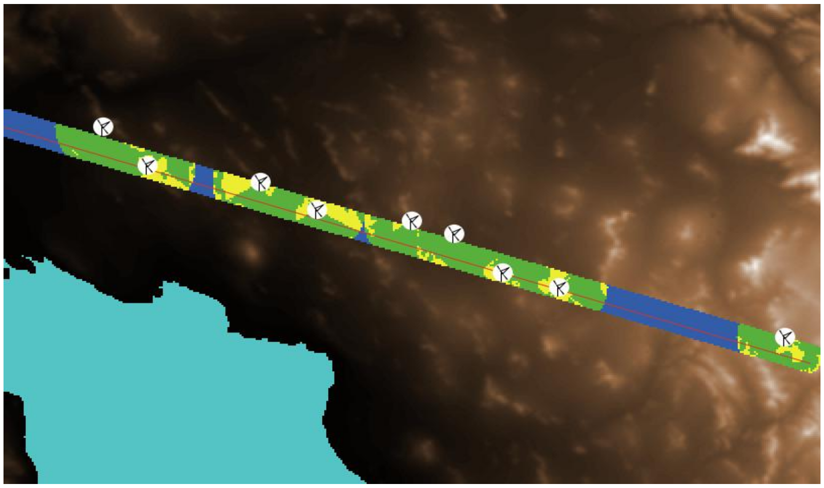


Figure 33: Architecture Coverage

The architecture coverage corresponds to a three-dimensional surface coverage that will be involved in the construction of the fitness function used to optimize the structure of the distributed detection system architectures for a specific CBP mission.

5.3.1.4. Types of Agents

Several types of agents may be modeled in the modeling and simulation environment described previously. These include:

- Ground items of interest, such as pedestrians, cars and trucks
- Aerial items of interest, such as general aviation aircraft, Unmanned Aerial Vehicles and Helicopters
- Maritime items of interest, such as speed boats, fishing boats, canoes and zodiacs
- Detection systems, such as radars, cameras, infrared goggles, acoustic sensors, chemical sensors and pressure sensors
- Customs and Border Protection units, such as CBP agents, CBP patrol units (mobile units), CBP command centers (headquarters)

In this research, the items of interest are mainly ground units trying to cross the Arizona-Sonora border, the detection systems are various types of radars and cameras, and the Customs and Border Protection units involve CBP agents and CBP patrol units. Specifics on the modeling and simulation of each of these agent types are described in subsequent sections.

5.3.2. Modeling and Simulation of Items of Interest

In the CBP mission scenario, the items of interest are modeled as agents moving in the theater of operations. They emanate uniformly from a random position in the southern part of the border. They are characterized by an average speed, which depends on the gradient of terrain elevation in their direction of travel, and an average radar cross section, which depends on the detection system involved in their detection. Both attribute values fall within the ranges defined in the sub-level morphological matrix for the "item of interest" given in Table 9. In general, items of interest tend to travel along paths of lowest elevations, exploring valleys or walking along water streams, to reach the border region. Nevertheless, some simulation scenarios may involve smarter items of interest that tend to avoid traveling along common low elevation paths where their risk of being detected is more likely, and that prefer evolving in mountainous or hilly regions as they make their way to the border.

In addition, some kind of intelligent behavior of the items of interest, including prior knowledge about the locations of deployed detection systems and/or about the performance of the currently deployed detection architecture and mobile patrol units, is captured via the concepts of "smuggling paths" and "exit points". As they make their way to the border, items of interest tend to travel towards pre-defined *smuggling paths* located in the northern part of the border. These *smuggling paths* are statistically high illegal activity areas where past successful illegal entries have occurred [462], [463]. They include highways and high-traffic roads where items of interest manage to mix in with legal activities and mislead more easily detection systems and CBP agents. Any number of smuggling paths may be defined interactively in the M&S environment to best represent the current operational situation. In this process, the user just needs to draw the structure of the smuggling paths directly on the "world" of the modeling and simulation environment with a mouse. Examples of pre-defined smuggling paths used in this research are depicted in Figure 34.

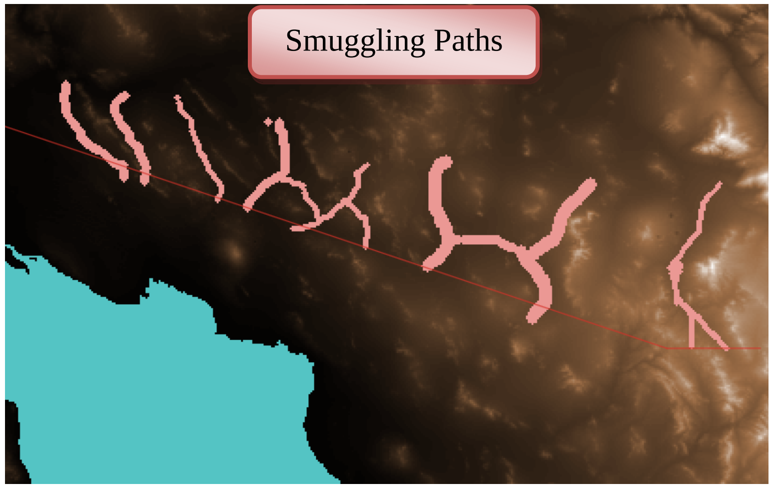


Figure 34: "Smuggling Paths" of the Items of Interest in the CBP Mission Scenario

If an item of interest manages to cross the border, even though it has been detected, it is assumed that it travels along these *smuggling paths* and is heading towards statically pre-defined *exit points* [462],[463]. The pre-defined exit points are characterized by an attractive potential corresponding to major U.S. cities in Arizona, such as Yuma, Phoenix, and Tucson. Any number of exit points may be defined by the user to best represent the current operational situation. In this process, the user just needs to create a text file specifying the number of exit points, and the corresponding latitudinal and longitudinal coordinates, as exemplified in Figure 35.

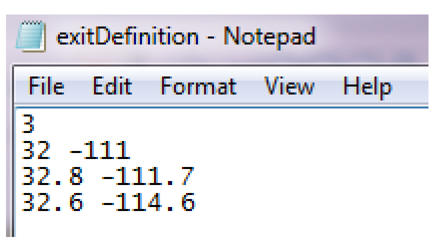


Figure 35: Example of a File Defining Exit Points

Examples of pre-defined exit points used in this research are depicted in Figure 36.

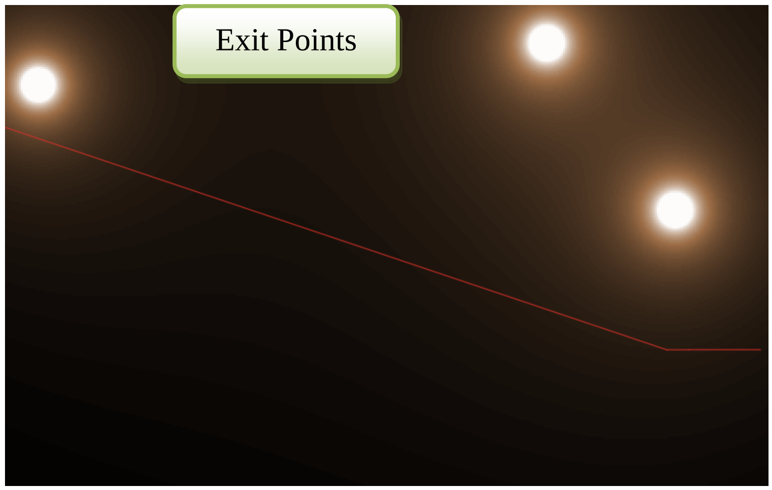


Figure 36: "Exit Points" for the Items of Interest in the CBP Mission Scenario

As they move in the theater of operations, items of interest first sample the few terrain patches located within a given distance from their current position, and select the one that is the closest to a pre-defined smuggling path and/or in the direction of an exit point. Once the items of interest reach their new position, a decision is made as to their next move. If an item of interest has not been intercepted by a CBP agent before it is within two minutes or 1 km of an exit point, it is assumed that it has escaped and has successfully penetrated the American territory illegally. In other words, it can no longer be distinguished from the rest of the population. In this case, the item of interest is removed from the simulation environment and a new item of interest of the same kind is created. This enables to keep a constant number of items of interest evolving in the modeling and simulation environment, and thus to have more leverage on the execution speed of the simulation and the repartition of the types of items of interest.

5.3.3. Modeling and Simulation of Detection Systems

In this study, several types of radars and cameras are considered. Their number, types, and locations are optimized so as to get the detection architectures providing the maximum terrain coverage at the minimum cost. The two main objectives of the problem are thus the three-dimensional terrain coverage and the cost of the complete detection architecture [14]. In this context, the sensor systems are modeled by their average range of detection calculated from their ranges of detection for each specific type of item of interest, their field of view (FOV), and their average cost which is a multi-parameter function of their main design parameters. The three types of radars and the three types of cameras of increasing performance and cost considered in this research are defined in Table 12.

Table 12: Description of the Sensor Systems Considered in the CBP Mission

Sensor Properties / *Sensor Type*	**Average Range of detection (km)**	**Field Of View (FOV) (°)**	**Average Cost (k$)**
Low Cost Radar (LCR)	12	100	150-220
Medium Cost Radar (MCR)	21	120	Around 580
High Cost Radar (HCR)	26	120	650-850
Low Cost Camera (LCC)	5	120	Around 14
Medium Cost Camera (MCC)	10	120	Around 40
High Cost Camera (HCC)	15	120	Around 160

These systems can either look in a fixed direction or rotate over a 360° angle. In the case where the system has a fixed field of view, the direction in which it is pointing is given by an azimuth angle that can take any value between 0 and 360°. This pointing direction is the heading of the detection system.

On the one hand, it is assumed that radar systems are always able to detect items of interest, even if it is raining, or if the atmospheric visibility is reduced, or in the presence of clouds. In other words, radars are assumed to be relatively insensitive to the presence of clouds, rain, snow and fog.

On the other hand, the design cost of cameras depends on their optical diameter and on their category, as expressed in Equation 4. Assuming average optical diameters provided in Table 11, one gets the average costs summarized in Table 12.

$$\text{design cost of LCC} = 10 + \left(\frac{\text{optical diameter}}{30}\right)^2$$

$$\text{design cost of MCC} = 30 + 1.5 * \left(\frac{\text{optical diameter}}{30}\right)^2$$

$$\text{design cost of HCC} = 120 + 2 * \left(\frac{\text{optical diameter}}{30}\right)^2$$

Equation 4

Finally, it is assumed that the design detection range of cameras is divided by two [460] in the presence of low altitude clouds, rain, snow and fog, or when the atmospheric visibility is reduced. In addition, it is assumed that both radars and cameras have infrared detection capabilities and have similar daytime and nighttime performance.

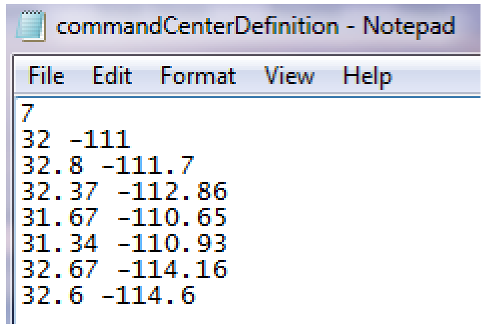
The role of detection systems is to detect items of interest as they intend to cross the border. In order to do so, a “detection” algorithm, based on the principles involved in the “line-of-sight” algorithm, has been developed. To start with, it is assumed that detection systems having a rotating field-of-view will eventually detect items of interest over the course of one rotation. Indeed, the speeds at which antennas of detection systems are rotating are much larger that the speeds at which items of interest are moving in the theater of operations. For detection systems having a fixed field-of-view, the “detection” algorithm directly applies. The “detection” algorithm assumes that items of interest moving in the theater of operations are confounded with their current patch location in the modeling and simulation environment. The “detection” algorithm further assumes that items of interest have negligible heights compared to the elevation value of the patch they are located at. They are therefore associated with a zero elevation value. In a first step, the "detection" algorithm determines which and how many items of interest are located within the base of the half-sphere of detection of the sensor by comparing their current latitudinal and longitudinal positions with the location of the center and the extent of the base of the half-sphere of detection. This provides a set *I* of patches occupied by items of interest within the half-sphere of detection of the sensor. In a second step, the "detection" algorithm changes the elevation values of the terrain patches in set *I* to zero, corresponding to the virtual height of the items of interest they represent. Using this information, the "line-of-sight" algorithm is then applied to each patch of the set *I* so as to determine which and how many of them are in the line-of-sight of the detection system. Relating back the resulting patches to their associated items of interest, this gives the set of items of interest that are detected by the detection system.

5.3.4. Modeling and Simulation of Customs and Border Protection Agents

In the CBP mission scenario, two additional categories of agents need to be considered. These are the Customs and Border Protection Agents and Patrol Units.

5.3.4.1. <u>CBP Agents</u>

The Customs and Border Protection agents are initially randomly assigned to CBP command centers located near the border. Any number of CBP command centers may be defined by the user to best represent the current operational situation. In this process, the user just needs to create a text file specifying the number of CBP command center, and the corresponding latitudinal and longitudinal coordinates, as exemplified in Figure 37. The corresponding CBP command centers are depicted in Figure 39 as blue houses.




```
7
32 -111
32.8 -111.7
32.37 -112.86
31.67 -110.65
31.34 -110.93
32.67 -114.16
32.6 -114.6
```

Figure 37: Example of a File Defining Customs and Border Protection Command Centers

The CBP agents have the ability to intercept potential illegal items of interest that have been detected near the border region. They are modeled as individuals moving at a random speed dependent on the gradient of terrain elevation in their direction of motion and on the properties of the items of interest they are trying to intercept, if any. The interception procedure unfolds as follows:

- A sensor system detects an item of interest and establishes a “detection link” with this item.
- The detection system starts tracking the item of interest and sends a “request for interception link” to the closest command center, which analyzes the status of the detected item of interest. If the item of interest is identified as being illegal or potentially harmful, the requested command center dispatches the closest available CBP agent (if any). It is assumed that a suspicious item of interest can only be intercepted by a single CBP agent at a time.
    - If no CBP agent is available for intercepting the item of interest for more than five minutes of real simulation time, then the item of interest is released free.
    - If a CBP agent is available for intercepting the suspicious item of interest, an “interception link” is created between the item of interest and the CBP agent assigned to it interception. It is assumed that the CBP agent receives updated information about the current speed and position of the suspicious item of interest as long as tracking by the detection system is effective. In

this case, the CBP agent moves along the shortest path to the item of interest, at a speed larger than the tracked item of interest.

- If tracking of the item of interest is lost, it is assumed that the CBP agent moves in a straight line from its last updated position and gives up pursuit of the item of interest if tracking is lost for more than five minutes of real simulation time.
- When the CBP agent and the suspicious item of interest get sufficiently close to each other, at a distance shorter than that traveled by the CBP agent over one simulation step, the interception is considered successful. The item of interest is then removed from the simulation environment and its statistics are stored (identity of the detection system that detected it, identity of the terrain patch where it was detected, times at which it crossed the border, location at which it crossed the border, identity of the CBP agent that intercepted it, time required for interception, identity of the patch where it was intercepted).
- Once the interception is complete, the CBP agent returns to its command center of origin at a speed dependent on the gradient of terrain elevation in its direction of motion.

5.3.4.2. CBP Patrol Units

They represent the CBP patrol agents patrolling the border with horses or cars. They are moving between pre-defined “patrol points” at a random speed. They have the ability to set up “mobile detection systems” as they patrol the border. This includes installing the detection device on the ground or in their truck, or preparing visible or infrared goggles for observation, stationing at the patrol point to monitor the border region for a certain amount of time, and packing up the surveillance equipment to horseback ride or drive to the next patrol point. Any number of patrol points may be defined by the user to best represent the current operational situation. In this process, the user just needs to create a text file specifying the number of patrol points, and the corresponding latitudinal and longitudinal coordinates, as exemplified in Figure 39. The corresponding CBP patrol points are depicted in Figure 39 as blue flags.

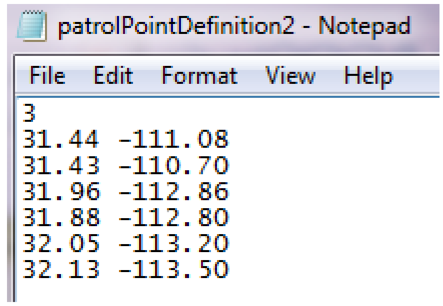


Figure 38: Example of a File Defining Patrol Points

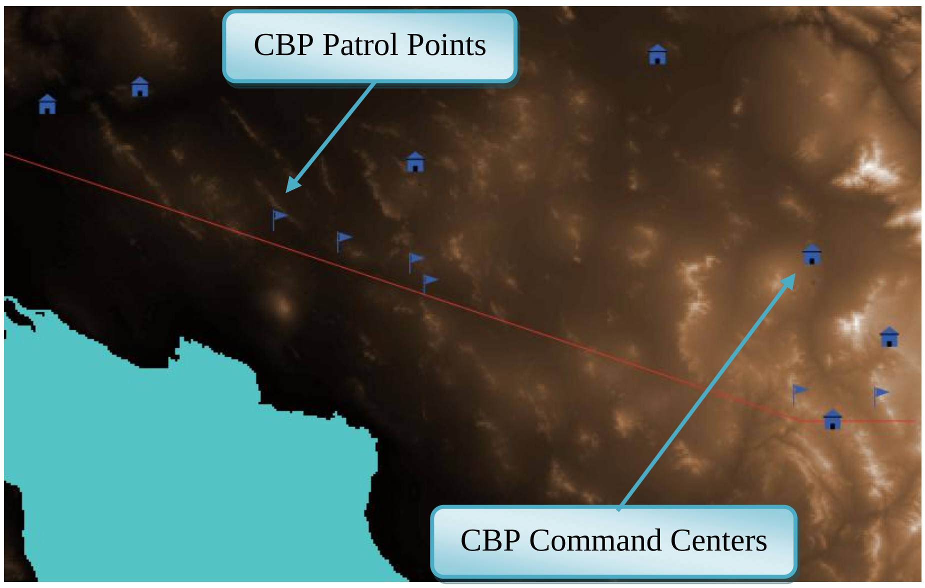


Figure 39: "CBP Command Centers" and "CBP Patrol Points" in the CBP Mission Scenario

5.3.5. Modeling of Weather, and Day and Night Conditions

In the CBP mission scenario, weather is modeled very simply. It is assumed that it is either sunny or rainy during the simulation. Indeed, although radars are relatively insensitive to weather conditions, the detection performance of cameras is highly dependent on the atmospheric temperature and humidity. Nevertheless, the relationship between the design detection range of cameras and the atmospheric visibility is not trivial and cannot be expressed as a linear law or any other simple relationship. Therefore and for practical purposes, the weather conditions in the theater of operations are modeled via the notion of atmospheric visibility. This assumption is valid in the sense that cameras are essentially sensitive to the presence of rain, and it was previously assumed that their design detection range was divided by two under rainy conditions. As the simulation progresses, the atmospheric visibility adapts to the time of the day and to the local climatic conditions (occurrence of rain, cloudy, or sunny) based on statistical weather data in the Arizona-Sonora border region.

Concerning modeling of day and night conditions, it was assumed that both radars and cameras have infrared detection capabilities and have similar daytime and nighttime performance. In addition, it assumed that day and night are equally spread over the 24-hour period, meaning that there is 12 hours of daylight and 12 hours of night.

5.3.6. Reducing the Problem Dimension: Selection of Most Promising Locations

In the optimization problem considered in this research, the numbers, types, and locations of surveillance systems are unknown and have to be decided simultaneously. More specifically, the problem has a large and multimodal search space, is combinatorial, with a mix of discrete and continuous variables having a large number of settings or large domains of variations. Therefore, it may be computationally intensive for optimization algorithms to find a solution to the above problem formulation. The most demanding part of the optimization resides in determining the locations of the sensor systems in the detection architectures from the complete spectrum of 10,700 positions available in the crossing zone inside the band of detection. Indeed, as mentioned previously, the sensor systems can have a longitudinal location ranging from 110.5° West to 114.8° West and a latitudinal position varying by 0.15° from its value at the border depending on the longitude. Among all the potential locations within the crossing zone, some may result in efficient coverage while some may not be worth looking at. For instance, a detection system located on top of a mountain is typically able to cover a significant portion of the terrain within its range of detection. On the contrary, a sensor located in a valley will most likely cover only a tiny portion of the terrain within its range of detection due to obstacles such as hills and mountains that are blocking its line-of-sight. In order to alleviate the work of the optimization algorithms, some promising positions, from where a relatively large portion of the surrounding terrain is visible, can be derived from the modeling environment for each type of detection system. To do so, each type of sensor is successively located at each of the 10,700 terrain patches in the crossing zone inside the band of detection depicted in Figure 29. Then, the number of patches within the detection range of the sensor and visible from the current sensor location is determined according to the terrain features. Finally, using the “line-of-sight” algorithm and the coverage efficiency defined in Equation 2 for the type of detection system considered (i.e. the ratio of the total number of patches within the range of detection of the sensor system to the number of patches actually seen by the sensor), a location- and sensor-specific detection performance is calculated and associated to the current location of the type of sensor considered. All of the possible locations in the crossing zone are evaluated this way for each type of detection system and compared to determine a set of most promising locations having the highest associated detection performance for the type of system considered. The sets of most promising locations for the three types of radar systems are depicted in Figure 40.

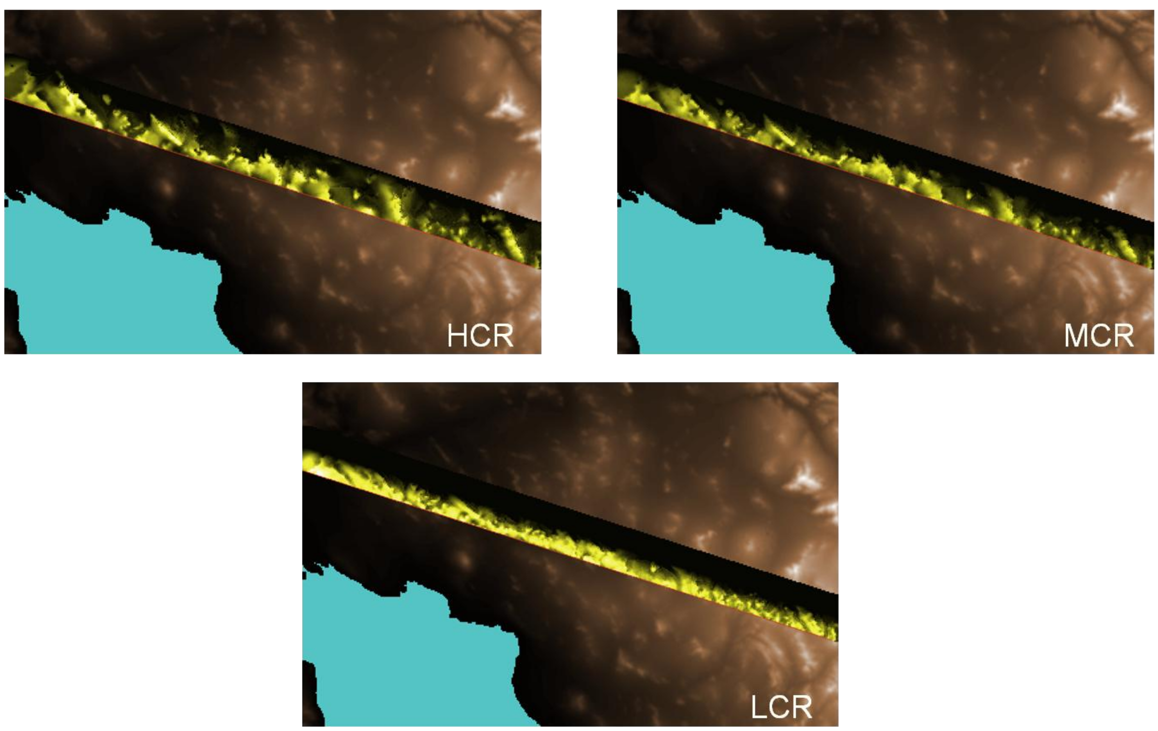


Figure 40: Sets of Promising Locations for the Radar Detection Systems in the CBP Mission Scenario

In Figure 40, the increasing yellow gradient of color models locations with increasing detection performance. Looking at the terrain's features in the band of detection (Figure 26), one may notice that the most promising locations are mainly located at higher elevations. Furthermore, lower ranges of detection translate into positions closer to the border (LCR compared to MCR compared to HCR), while larger ranges of detection lead to more distinct positions (HCR compared to MCR compared to LCR). One may also note that the sets of most promising locations are very similar for the three radar detection systems represented in Figure 40. In fact, it can be predicted that the most promising locations for sensor systems with large detection ranges will also be most promising locations for sensor systems with lower detection ranges. These locations thus constitute a global set of most promising positions common for all types of sensors. Then, the lower the detection range, the more the number of promising locations are added to the global common set. Indeed, detection systems with lower detection ranges are globally less impacted by terrain obstacles and thus are generally able to cover most of the terrain within their ranges at any given location compared to sensor systems with larger detection ranges. In particular, they will be more efficient in valleys. Finally, the sets of most promising locations for the camera detection systems are highly similar to those depicted in Figure 40 for the radar detection systems, and the above comments concerning the structure of the most promising positions for cameras of decreasing ranges of detection also applies.

Once the sets of most promising locations for each type of detection system are obtained, they can be combined to form a global set of most promising locations where sensor systems may be placed to compose the sought-after detection architectures. The combination is such that the final set of most promising locations covers a relatively wide spectrum of detection ratios above a certain threshold and is not limited to only those positions having the largest detection ratios. Indeed, when combining systems to create a detection architecture, all the sensor systems cannot be placed at the same best locations. Some have to be placed at poorer positions. This significantly decreases the number of locations to look at and ensures that the sensor systems are only placed at "efficient" positions providing the highest coverage ratio. A set of 256 most promising locations has been defined in this study so that each location can be represented as an integer from 1 to 256. This makes the encoding of the location of a sensor system easier, especially for the modified genetic algorithm to be described later. The 256 most promising locations are summarized in Appendix I and displayed in Figure 41. In Appendix I, X and Y are the coordinates in the NetLogo environment.

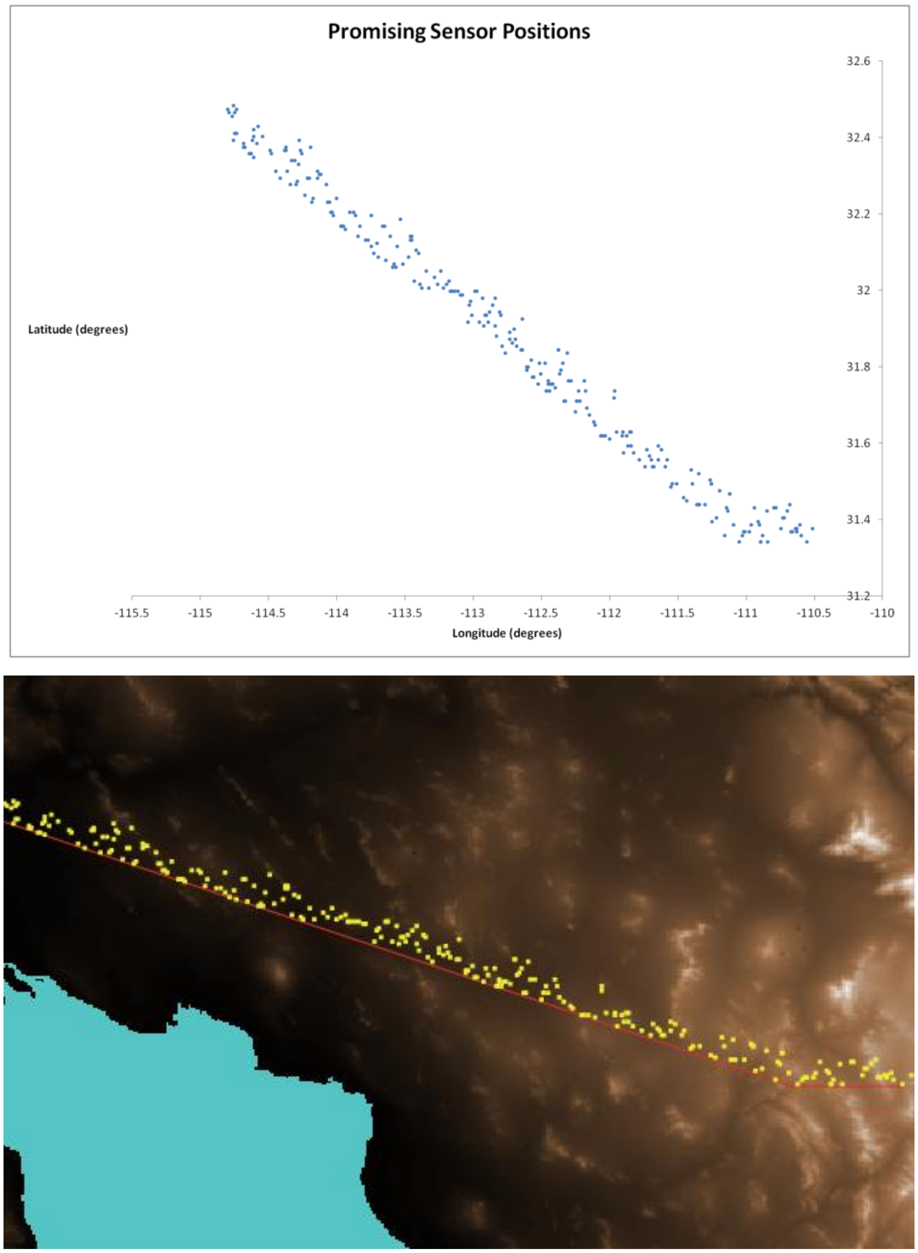


Figure 41: Plot of the 256 Most Promising Positions

5.3.7. Simulation Environment

Figure 42 is an example simulation of a notional detection architecture involving mobile patrol units, in the NetLogo environment. The detection architecture is defined from a text file which contains the total number of detection systems in the detection architecture, and the same number of lines specifying the properties of the constituting detection systems, namely their latitudinal and longitudinal coordinates, their heading, their type and their rotational state (rotating or fixed field-of-view). The first few lines of the text file defining the detection architecture depicted in Figure 42 is exemplified in Figure 43.

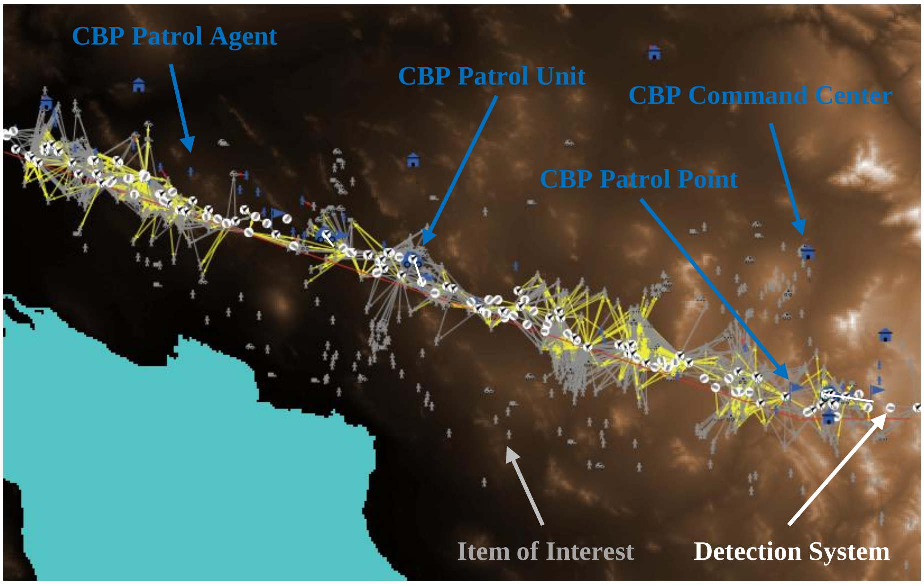
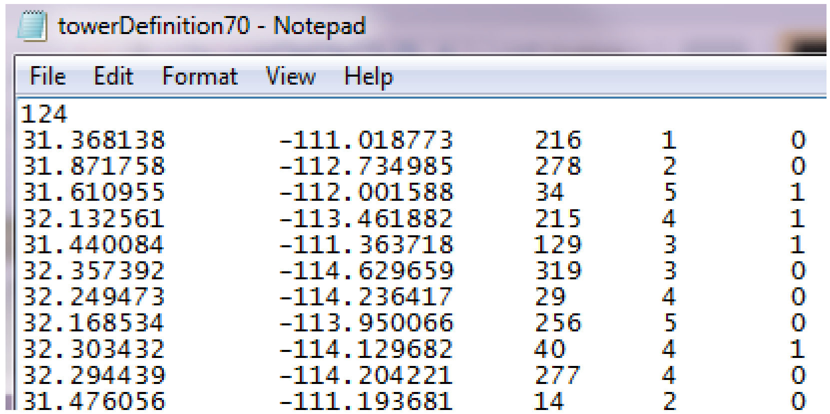


Figure 42: Simulation of a Notional Detection Architecture in the NetLogo Environment



towerDefinition70 - Notepad

File Edit Format View Help

124

31.368138	-111.018773	216	1	0
31.871758	-112.734985	278	2	0
31.610955	-112.001588	34	5	1
32.132561	-113.461882	215	4	1
31.440084	-111.363718	129	3	1
32.357392	-114.629659	319	3	0
32.249473	-114.236417	29	4	0
32.168534	-113.950066	256	5	0
32.303432	-114.129682	40	4	1
32.294439	-114.204221	277	4	0
31.476056	-111.193681	14	2	0

Figure 43: Example of a File Defining a Detection Architecture

In the simulation environment, the blue persons are the CBP agents, the grey icons are the items of interest (pedestrians, cars, and trucks), the blue houses are the CBP command centers, the blue cars are the CBP mobile units, and the blue flags are the CBP patrol points. The yellow lines model links between detected items of interest and sensor systems that have detected them, the grey lines represent links between sensor systems and items of interest within their detection ranges but out-of-sight, the red lines correspond to links between detected items of interest and CBP agents assigned to their interception, and the white lines model links between a mobile patrol unit and the patrol point it is travelling to. Links may be turned on or off by the user as the simulation goes using vertical switches. This is shown in Figure 44.

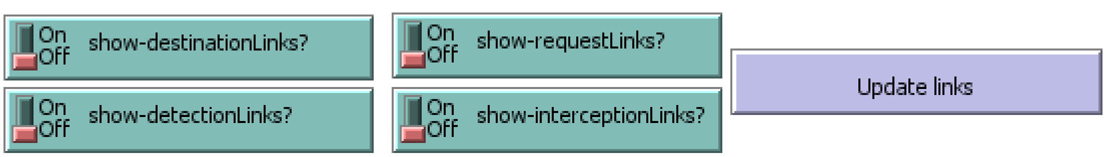


Figure 44: Simulation Environment – Links

A simulation step corresponds to a minute of real time. Therefore, agents moving in the simulation environment represent real agents in the theater of operations whose state (including position, detection information, interception information, etc) is updated every minute.

Several initial conditions of the simulation may be tuned by the user via slide bars, as depicted in Figure 45. These are divided in three main categories defining simulation properties of the items of interest, the CBP agents, and the mobile patrol units. For instance:

- "communicationTime" sets to the time required for the detection systems having detected an item of interest to send the detection information over to the closest CBP command center, and then from the corresponding CBP command center to the CBP agent dispatched for its interception.
- "CBPpatrolRadius" corresponds to the radius of interception of items of interest by CBP agents. An item of interest that has previously been detected by a detection system and that is currently being tracked by the same detection system, has to be located within this distance of an available CBP agent for the agent to be dispatched for its interception.
- "setupCBPdepth" represents the maximum distance between the border and the initial position of a CBP agent.
- "installationTime" sets to the time required for the mobile patrol units to install their detection systems on their mobile platforms. It is about 5 minutes in the example provided.
- "stationTime" corresponds to the period of time over which the surrounding environment is scanned by the mobile detection systems once installed. It is about 17 minutes in the example provided.
- "deinstallationTime" represents the time required for the mobile patrol units to dismount their detection systems from their mobile platforms before moving on to the next patrol point. It is about 9 minutes in the example provided.

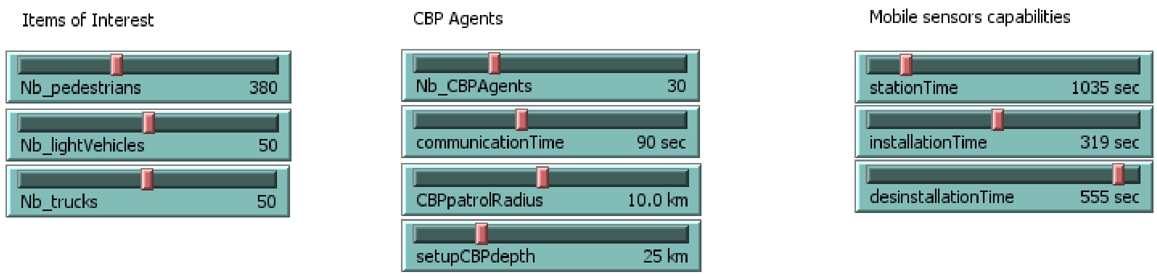


Figure 45: Simulation Environment – Initial Conditions

In terms of agent statistics, it is assumed that every time an item of interest is removed from the simulation environment because it has reached an exit point, or it has been intercepted by a CBP agent, a new item of interest of the same kind is created in the southern part of the border within 50 km. This enables to keep a constant number of items of interest evolving in the modeling and simulation environment, and thus to have more leverage on the execution speed of the simulation and the repartition of the types of items of interest. On the other side, CBP agents and mobile units, if any, are neither destroyed nor replaced, meaning that the numbers of CBP agents and mobile units specified as part of the initial conditions, remain constant during the simulation.

Several output parameters are tracked during the simulation to determine the operational performance of the detection architecture of interest. These are:

- The number of items of interest that have crossed the border (detected or undetected), named "IoIcrossed".
- The number of items of interest that have successfully been intercepted by a CBP agent, denoted "IoIintercepted".
- The number of items of interest that have reached an exit point, called "IoIescaped". These items of interest may have been detected, and a CBP agent may have tried to intercept them.
- The number of items of interest that have been detected, named "IoIDetected".

The output parameter values are updated constantly as the simulation progresses and written in output boxes, as those shown in Figure 46.

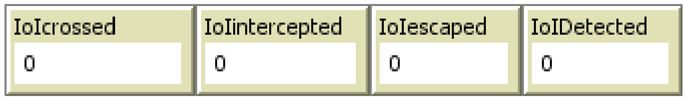


Figure 46: Simulation Environment – Output Parameters

To conclude, various data may be stored during the simulation to create statistics. This includes, but is not limited to:

- The trajectory of the ground items of interest
- The identities of the detection systems that have detected items of interest
- The identities of detected items of interest
- The identities of the terrain patches where items of interest were detected
- The times at which items of interest crossed the border
- The location at which items of interest crossed the border
- The identities of the CBP agents that intercepted items of interest
- The times required for each interception
- The identities of the terrain patches where items of interest were intercepted
- The total number of items of interest generated during the simulation, which is the sum of the number of items of interest present in the simulation at the current step, the number of items of interest that have escaped (reached an exit point) so far, and the number of items of interest that have been detected so far

5.3.8. Capabilities of the Modeling and Simulation Environment

The modeling and simulation (M&S) environment developed for this research is able to accommodate any kind of homeland security mission taking place in any kind of topographic, climatic, and operational environment. Indeed, the M&S framework has the capability to model any type of elevation, climatic, and operational data of interest to the decision maker. It represents a “plug and play” environment in which any number and any type of agent may be defined at leisure to model and simulate specific situations.

Although the present study focuses on the detection of ground items of interest as they are trying to cross a land border, any type of ground, aerial, or maritime items of interest may also be defined. Similarly, depending on the homeland security mission considered, any type of sensor device, such as visible sensors, infrared sensors, acoustic sensors, chemical sensors, and pressure sensors may be incorporated in the M&S environment. It is also possible to introduce some nuance in the intelligent behavior of the items of interest, such as a priori knowledge about how to avoid detection and interception, spoofing, jamming, and non-cooperative smart behavior. In this context, the M&S environment developed as part of this research may be used for various types of analyzes.

The present study focuses on the optimization of distributed detection system architectures to monitor a border region for illegal crossings, and on the performance analysis of the resulting detection architectures. However, the M&S framework has the ability to play any kind of “what-if” scenario that may be used for instance to:

- Gain insight into an operational situation
- Train military personnel to respond to specific operational conditions when performing on the terrain
- Demonstrate the capabilities of sensor systems to potential customers
- Design of new sensor systems adapted to a specific situation
- Analyze the internal relationships and compatibilities between the various agents modeled
- Adapt existing surveillance architectures to changing operational conditions
- Enhance the operational effectiveness of existing protective solutions
- Analyze the sensitivity of the performance and the cost of detection architectures to changes in their composition or in the operational situation

5.3.9. Concluding Remarks on Modeling and Simulation of the Customs and Border Protection Scenario

The previous sections have demonstrated how a physics-based approach combined with agent-based modeling provides a means to develop a modeling and simulation environment for the design, modeling, and simulation of distributed detection system architectures in the context of homeland security. Such modeling capabilities, complemented by the multi-level morphological analysis method constructed in previous sections, adds to the definition of a structured, robust, rigorous and traceable process for the simulation of notional homeland security mission scenarios. The aforementioned sections addressed the modeling, simulation, and optimization environment research question as well as the modeling and simulation research question, and validated the corresponding hypotheses.

CHAPTER VI

IMPLEMENTATION – OPTIMIZATION OF THE DETECTION ARCHITECTURE FOR A PROOF-OF-CONCEPT SCENARIO

The optimization of a portfolio of detection architectures for the CBP mission scenario implies determining the combinations of the numbers, types, and positions of distributed detection systems able to provide the maximum detection coverage at the minimum cost. The **optimization problem** may therefore be **characterized** as follows:

- Mixed (involving continuous and discrete variables)
- Combinatorial
- Involving a large,
- Multimodal,
- And non-smooth search space

The employment of an **optimization algorithm** presenting the following **properties** is therefore required:

1. Non-linear
2. Adaptive
3. Multi-criteria
4. Able to handle both continuous and discrete variables

Furthermore, the goal of the CBP mission scenario is to gain insight into the real world problem without being overwhelmed by complex and tedious optimization tasks. The only requirement in this case is to find a **sufficiently good solution**. This implies finding a portfolio of detection architectures providing adequate detection performance at relatively low costs. In other words, the good-enough detection architectures are required to be able to adequately detect items of interest using affordable detection systems. In this context, the portfolio of detection architectures is obtained by optimizing the problem according to both **coverage performance** and **cost**. The performance of a detection architecture is calculated using the M&S framework NetLogo, as described in Equation 3, while the cost of the detection architecture is defined as the sum of the design costs of its component systems. Costs related to the deployment, exploitation, and maintenance of the detection architecture are not considered in this research. However, coverage and cost are **conflicting objectives** in the sensor placement problem since increasing the number of sensors in the architecture increases both coverage and cost. Multiple conflicting objectives call for **multi-objective evolutionary optimization** such as Genetic Algorithm and Particle Swarm Optimization which have proven to be well suited to tackle the problem of sensor placement under a variety of situations, in particular for border surveillance and intrusion detection.

6.1. Comparison, Selection, and Modification of Optimization Algorithms to Solve the Proof-of-Concept Scenario

6.1.1. Comparison and Selection of Optimization Algorithms

Based on the characteristics of the CBP optimization problem, a table can be created to compare the ability of various optimization algorithms to satisfy the selection criteria for the optimization problem. These algorithms were identified in the literature on optimization problems related to distributed sensor system locations. They include the Glowworm Swarm Optimization (GSO) [144], the Ant Colony Optimization (ACO) [145], the Particle Swarm Optimization [160],[169],[142],[146],[170], various types of Genetic Algorithms [171],[172],[168],[161],[162],[164],[155],[165],[156],[173],[166], the Simulated Annealing [170], Integer Linear Programming [148],[163], and Tabu Search [137]. A more exhaustive list of the main optimization algorithms used in distributed sensor system studies along with their specific application domains and primary objective functions are summarized in Table 13.

Table 13: Optimization Algorithms Used in Distributed Sensor Network Placement Problems

Field	Algorithm	Primary Objective	Reference
Surveillance	Integer Edge Covering	Maximize coverage	Bottino and Laurentini (2008) [139]
Wireless Sensor Network Deployment	Glowworm Swarm Optimization	Maximize coverage	Liao, Kao, and Li (2011) [144]
Surveillance	Immune-based two-Phase Approach	Minimize maximal failure detection probability	Hsieh et al. (2009) [140]
Wireless Sensor Network Deployment	Ant Colony Optimization	Maximize coverage	Liao, Kao, and Wu (2011) [145]
Sensor Placement	Integer-Coded Genetic Algorithm	Maximize Coverage	Boying and Xiankun (2011) [171]
Sensor Placement	Integer Programming	Maximize Coverage	Minjie et al. (2011) [163]
Camera Network	Particle Swarm Optimization	Maximize Coverage	Xu et al. (2011) [142]
Wireless Sensor Network Deployment	Multi-Objective Particle Swarm Optimization	Maximize Coverage and Network Lifetime	Pradhan and Panda (2012) [146]
Security Monitoring	Maximal Covering Location Problem + Visibility Analysis	Maximize Coverage	Murray et al. (2007) [137]
Transportation Management	Genetic Algorithm	Minimize Travel Time	Kim et al. (2010) [172]
Water Distribution Network Monitoring	Multi-Criteria Optimization	Minimize Detection Time and Contaminated Population	Krause et al. (2008) [138]
Wireless Sensor Network Deployment	Non-Linear Programming	Minimize Energy Consumption and Travel Distance, Maximize Coverage	Guerriero et al. (2011) [147]
Air Pollution Monitoring	Multi-Objective Optimization	Maximize Coverage	Trujillo-Ventura and Ellis (1991) [159]
Directional Sensor Network Deployment	Survey of Various Optimization Techniques (cf. Table 5 of the reference)	Maximize Coverage and Network Lifetime	Guvensan and Yavuz (2011) [151]
Damage Detection / Structural Health Monitoring	Particle Swarm Optimization	Maximize Coverage	Abdalla and Al-Khawaldeh (2012) [160]
Wireless Sensor Network Deployment	Integer Linear Programming	Minimize Energy Consumption and Maximize Network Lifetime	Prommak and Modhirun (2012) [148]

Table 13: Optimization Algorithms Used in Distributed Sensor Network Placement Problems (Continued)

Large Area Surveillance	Binary Optimization	Maximize Visibility	Erdem and Sclaroff (2004) [152]
Damage Characterization	Genetic Algorithm	Maximize Coverage	Yan et al. (2007) [161]
Sensor Placement	Particle Swarm Optimization	Maximize Coverage	Boying and Xiankun (2011) [169]
Fault Detection	Genetic Algorithm	Maximize Coverage	Worden and Burrows (2001) [162]
High-Rise Structural Health Monitoring	Genetic Algorithm	Improved Information Matrix	Zhan et al. (2012) [164]
Water Quality Monitoring	Genetic Algorithm	Maximize Detection of Contaminants and Minimize Contaminated Population	Telci et al. (2009) [155]
Structural Health Monitoring	Genetic Algorithm	Modal Assurance Criterion	Yang and Zhang (2012) [165]
Water Distribution System Monitoring	Non-Dominated Sorting Genetic Algorithm (NSGA-II)	Minimize Time Delay in Detection of Intrusion Events and Maximize Sensor Detection Redundancy	Shen and McBean (2011) [156]
Water Distribution Systems / Facilities Planning and Design	Multi-Objective Minimax Optimization (summary of other methods from other sources in Table 1 of the reference)	Minimize Volume of Contaminated Water and Detection Time, Maximize Coverage	Xu et al. (2010) [157]
Bridge Structural Health Monitoring	Single Parents Genetic Algorithm with Different Fitness Functions	Maximize Linear Independence, Minimize Energy Consumption	Han-bing et al. (2011) [166]
Distributed Sensor Network Deployment	Polynomial-Time Algorithms	Maximize Coverage under Imprecise Detections and Terrain Properties	Dhillon and Chakrabarty (2003) [136]
Urban Homeland Security	Combinatorial Optimization	Minimize Energy Consumption, Maximize Airborne Contaminants Detection, Minimize Affected Population	Hamel et al. (2010) [141]

Table 13: Optimization Algorithms Used in Distributed Sensor Network Placement Problems (Continued)

Diver Detection / Sensor Placement	Stochastic Optimization	Maximize Detection, Minimize Energy Consumption	Molyboha and Zabarankin (2012) [175]
Wireless Sensor Network Deployment	Survey of Various Optimization Techniques (cf Table 2 of the reference)	-	Younis and Akkaya (2008) [150]
Directional Sensor Network Deployment	Polynomial-Time Algorithm	Maximize Coverage, Minimize Energy Consumption and Total and Maximum Rotation Angles of Sensors	Tao et al. (2012) [143]
Objects Layout Optimization	Simulated Annealing and Particle Swarm Optimization	Minimize Container Radius and Mass Imbalance	Xiao et al. (2007) [170]

The results of the comparison and of the selection of optimization algorithms for the CBP optimization problem are depicted in Table 14.

Table 14: Comparison of Relevant Optimization Algorithms for the CBP Optimization Problem

Algorithms / ***Criteria***	**Glowworm Swarm**	**Ant Colony**	**Particle Swarm**	**Genetic Algorithm**	**Simulated Annealing**	**Integer Linear Programming**	**Tabu Search**
ABLE TO HANDLE OPTIMIZATION PROBLEMS WITH THE FOLLOWING PROPERTIES							
Mixed	✓	✓	✓	✓	✓	✓	⬇
Combinatorial	✓	✓	✓	✓	✓	✓	⬇
Large Search Space	✓	✓	✓	✓	✓	✓	⬇
Multimodal Search Space	✓	✓	✓	✓	✓	✓	✓
Non-Smooth Search Space	✓	✓	✓	✓	✓	⬇	⬇
Multi-Criteria / Conflicting Objectives	✓	✓	✓	✓	✓	⬇	⬇
PRESENTING THE FOLLOWING CHARACTERISTICS							
Fast	✓	✓	✓	✓	✓	✓	⬇
Easy to Implement	⬇	⬇	✓	✓	⬇	⬇	⬇
Flexible / Adaptive	✓	✓	✓	✓	✓	⬇	✓
Versatile	✓	✓	✓	✓	✓	⬇	⬇

Legend:

- ✓ satisfies property
- ⬇ is worse than other options

As can be derived from Table 14, Genetic Algorithm and Particle Swarm Optimization seem the most appropriate approaches to solve the identified problem. These evolutionary algorithms have been modified from their original versions in order to take into account the main characteristics of the optimization problem under study and to enhance their performance at finding a proper solution. The **modified Genetic Algorithm** and the **modified Particle Swarm Optimization Algorithm** developed for the purpose of the CBP optimization problem are described in subsequent sections.

6.1.2. Modification of Genetic Algorithm and Particle Swarm Optimization

Evolutionary algorithms are powerful and computationally reasonable optimization techniques that mimic natural selection and survival of the fittest to find solutions in an unknown search space. The procedure of optimization is based on a population of potential candidate solutions that is refined according to a fitness function. The candidate solutions are either chromosomes encoded in genes in the Genetic Algorithm or agents (birds, fishes, etc) encoded in body parts in the Particle Swarm Optimization Algorithm. Chromosomes and agents are commonly called individuals. Genes and body parts are called components. Components specify the attributes of the solutions, namely the variables of the problem, and determine the fitness of the solution. In a typical optimization problem, each individual is composed of a number of components equal to the number of variables in the problem. In the case of the CBP optimization problem, this means that each individual would represent a single sensor system and would be made out of four components characterizing the four variables of the problem:

- The location of the sensor system (longitude and latitude), which is represented as an integer between 1 and 256
- The state of the sensor system, either rotating (1) or non-rotating (0)
- The orientation of the sensor system (direction in which the sensor system is looking if it has a fixed field-of-view), which ranges from 0° to 360° and is defined from a latitudinal line in the NetLogo world
- The type of the sensor system, which ranges from 0 to 7 (0 and 7 means that the sensor is not active, 1 to 6 represent the LCR, MCR, HCR, LCC, MCC, and HCC respectively). The value of 7 is added for encoding purposes (see next subsection)

If one were to implement the traditional structure of a chromosome in the GA or of a particle in the PSO for the CBP optimization problem, then an individual in the population would represent a single detection system, as notionally depicted in Figure 47.

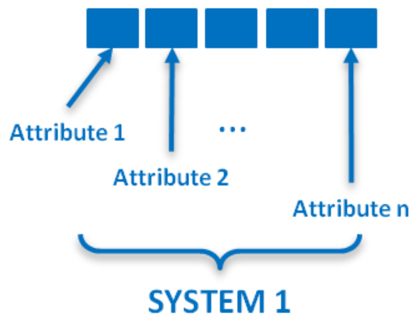


Figure 47: Structure of Chromosomes and Particles in Traditional Versions of the Genetic Algorithm and Particle Swarm Optimization Algorithm

However, in this research, the goal is to optimize the combination of various types of detection systems over large terrains so that they jointly provide a maximum performance at the minimum cost. The detection systems composing the resulting detection architecture are not working independently but rather interact with each other and combine their performance into a global capability. This is the main principle behind the emergent behavior of systems-of-systems. In this context, the optimization process involves optimizing not only the types and the locations of the sensor systems constituting the surveillance architecture but also, and more importantly, the total number of systems in the final architecture. This implies that each individual in the optimization algorithms should not represent a single detection system but rather the complete surveillance architecture.

The structure of chromosomes and particles in traditional versions of the GA and PSO algorithm is thus modified to represent a detection architecture rather than a single sensor system, as required in the CBP optimization problem. In the adapted versions of the GA and PSO algorithm, each individual is composed of a number of parts equal to the total number of systems in the detection architecture, and each part is itself composed of a number of sub-parts equal to the number of attributes or variables characterizing the corresponding sensor system. In order to implement this complex optimization approach, one may set a threshold on the total number of detection systems in the final detection architecture. This maximum number of detection systems allowed in the final detection architecture is denoted *S_max*. Then, each individual is composed of *S_max* parts, themselves composed of four sub-parts, as depicted in Figure 48. In this study, *S_max* may vary from 10 to 200 with a step of 10 according to published literature on the topic [462],[463].

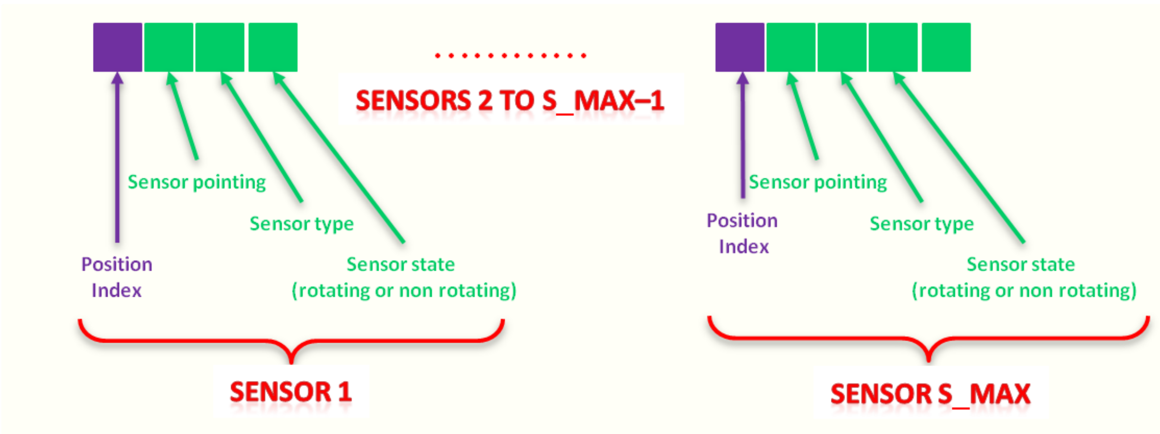


Figure 48: Structure of Chromosomes and Particles in the Adapted Versions of the Genetic Algorithm and Particle Swarm Optimization Algorithm

Since the number of sensor systems in the final detection architecture is to be determined as part of the optimization process, some individuals may have empty parts, meaning that the corresponding system is not active and is not included in the calculation of the fitness function for the corresponding detection architecture.

6.1.3. Construction of an Objective Function

As mentioned earlier, the fitness function or objective function to be optimized is composed of two components:

- A performance measure: the surface covered by the detection architecture in the band of protection, defined in Equation 3

- A cost measure: the sum of the design costs of the sensor systems composing the detection architecture

However, surface coverage and cost are two conflicting objectives, not only in terms of optimization but also in the minds of decision makers. Indeed, different decision makers may value surface coverage and cost differently. The relative balance between both objectives may also depend on the problem to be solved or on the operational context. In any case, the objective function to be optimized needs to capture such preferences. This is done through a parametric study. A power law is introduced to create an aggregated objective function. In this study, the detection architecture coverage is to be maximized with respect to the corresponding cost. Hence, the objective function can be described by Equation 5.

$$\frac{(\text{Architecture Coverage})^{\alpha}}{k^{*}\text{Architecture Cost}}$$

Equation 5

In order for the objective function to be properly scaled, the architecture coverage and the architecture cost need to be of the same order of magnitude. Surface coverage is expressed as a percentage and is therefore between 1% [4] and 100%. Therefore, the architecture cost, which is initially expressed in thousands of dollars, must be scaled back to a value between 1 and 100 as well. This is done by scaling the real architecture cost by an appropriate factor k, here 10^{-2}. The weight factor α may then be varied to sweep across a wide range of CBP mission scenarios. Cases for which $\alpha > 1$ are representative of situations where coverage is given less weight than cost, while cases where $\alpha < 1$ correspond to situations where coverage is given more weight than cost. In the subsequent analysis, $\alpha = 1$ unless otherwise specified.

[4] Cases where the detection coverage is between 0% and 1% are highly improbable. There will always be at least one sensor system in the architecture such that its coverage will always be strictly larger than 1% even with a low cost camera.

The modified versions of Genetic Algorithm and Particle Swarm Optimization described in the previous section have been implemented in Matlab. Each optimization search starts with an initial random population of detection architectures. At each generation, the cost of each individual in the population is calculated. The population of detection architectures is sent next to the modeling environment to determine the surface coverage of each of its members as per Equation 3, taking into account the potential overlap in coverage from neighboring detection systems and terrain obstacles. The resulting surface coverage and associated cost are then combined to determine the fitness function value for each detection architecture in the population. Finally, a new population is generated according to the specific reproduction procedures of each of the optimization algorithms considered.

6.1.4. Features Specific to the Modified Genetic Algorithm

In the genetic algorithm, the architectures having the largest fitness, i.e. the largest coverage to cost ratio as given by Equation 5, are selected to reproduce. The reproduction process is based on a modified proportional representation with elitist approach in which one or more of the best architectures in the current population is/are directly passed to the intermediate generation without being crossed-over with another architecture. The other elements of the intermediate generation are obtained by breeding between pairs of architectures in the parent population to create offsprings. A two-point crossover scheme is adopted. The architectures in the intermediate population are then uniformly mutated to generate the next generation on which the same process described above is successively applied until convergence is reached. A uniform mutation scheme in which each bit of each detection architecture in the population is flipped according to the mutation probability is adopted. The modified genetic algorithm is assumed to have converged whenever the objective function has not improved by more than 10^{-8} over a pre-defined number of successive generations denoted *StopIte*. However, in order to avoid stagnation of the GA, a maximum number of generations of 300 is specified. If this number is reached, then the GA is stopped and other parameter settings must be determined so that the GA can converge to a solution.

Each architecture is encoded in Gray coding and the resolution on the variables is a function of their character, discrete or continuous, and of their range of variation. More precisely, the location of a sensor which encompasses both its longitudinal and its latitudinal locations in an integer ranging discretely from 1 to 256 (number of most promising locations chosen) is encoded in eight bits ($2^8 = 256$). The discrete rotating/non-rotating state of the sensor system which can be either 1 (rotating) or 0 (non-rotating) is encoded in one bit ($2^1 = 2$). The heading of the sensor system which ranges continuously from 0° to 360° is discretized in 256 different states and encoded in 8 bits. Finally, the type of the sensor system which ranges discretely from 0 to 7 is encoded in 3 bits ($2^3 = 8$ is sufficient).

It is generally understood that GA parameters, including the crossover rate, the mutation rate, the population size, and the convergence criterion, are determined mainly by trial and error. The population size should be such that it is not too small in order for the GA to perform a wide search of the design space but also not too large in order for the GA to be computationally efficient. In addition, the mutation rate should not be too large to avoid random searches of the design space. On the contrary, large values for the crossover rate typically work better for a wide range of applications. In the present CBP mission application, the crossover rate is set to 70% based on similar sensor placement work, while the population size, the mutation rate and the second convergence criterion need to be investigated further as described in subsequent sections.

6.1.5. Features Specific to the Modified Particle Swarm Optimization

Similarly to the genetic algorithm, a particle swarm optimization algorithm evolves a population of agents or particles to search the design space of the problem. Each particle is characterized by a position vector and a velocity vector. The position vector in this case represents a potential detection architecture solution to the CBP mission problem, and the velocity vector models the distance in the design space traveled by the particle between two successive optimization steps. The basic optimization framework consists of three operations:

1. The generation of the particles' positions and velocities,
2. The update of the particles' velocities,
3. And the update of the particles' positions.

Each particle in the current population (or swarm) tracks its location in the design space by means of a vector that both accelerates the particle in the direction of the best position it has visited so far, and in the direction of the overall best position visited by the swarm so far defined as the best position among all the best positions of the particles in the swarm. The optimization procedure adopted in this study is as follows, where k represents the current iteration step:

1. A population of position vectors $\vec{x_i}^k$ is randomly generated, each one modeling a potential protection architecture solution to the problem:
   a. Each position vector is composed of a number of components equal to the maximum number of sensor systems allowed in the architecture (*S_max*)
   b. Each component is constituted of a number of sub-components equal to the number of variables in the problem
   c. Each sub-component has a value randomly defined from the range of variation of the variable it is representing
2. A population of velocity vectors $\vec{v_i}^k$ is randomly generated, where $\left\|\vec{v_i}^k\right\|$ is the step length for the update of $\vec{x_i}^k$

a. Each velocity vector is composed of a number of components equal to the maximum number of sensor systems allowed in the architecture (*S_max*)
b. Each component has a number of sub-components equal to the number of variables in the problem
c. Each sub-component has a value bounded in the range between the negative of a maximum velocity V_{max} and the positive of the same maximum velocity. In this study, the maximum velocity V_{max} is taken equal to four

3. The private best position of each particle is initialized to its initial position as $\vec{p_i}^k = \vec{x_i}^k$, where $\vec{p_i}^k$ stores the best position found in the design space by particle i during its history. The objective function of the problem provides the best position for each particle in the swarm
4. The global best position of the swarm is initialized to the best position among all the local best positions of the particles in the swarm as $\vec{g}^k = \max_i(\vec{p_i}^k)$
5. While the stopping criteria is not satisfied, the following procedures are executed:
   a. For each particle in the swarm, the velocity vector is updated by Equation 6:

$$\vec{v_i}^{k+1} = w\vec{v_i}^k + C_1 rand \frac{\vec{p_i}^k - \vec{x_i}^k}{\Delta t} + C_2 rand \frac{\vec{g}^k - \vec{x_i}^k}{\Delta t}$$

Equation 6

Where w is the inertia factor which typically ranges from 0.4 to 1.4 **[464]**, C_1 is the self confidence factor which often ranges from 1.5 to 2 **[464]**, C_2 is the swarm confidence factor which generally ranges from 2 to 2.5 **[464]**, and Δt is the optimization time step. In Equation 6, $rand \frac{\vec{p_i}^k - \vec{x_i}^k}{\Delta t}$ models the influence of the particle's ability to remember the best location it has visited so far on its next move, while $rand \frac{\vec{g}^k - \vec{x_i}^k}{\Delta t}$ models the influence of the best location visited by the swarm so far on the next move of particle *i*. The original PSO algorithm **[464]** uses the values of 1, 2, and 2 for *w*, C_1, and C_2 respectively. In the modified PSO considered in this paper, the inertia factor *w* linearly decreases from 0.9 to 0.4 to model the decreasing influence of past velocity as the optimization progresses, the self confidence factor C_1 is taken equal to 2, while the swarm confidence factor C_2 can vary from 2 to 2.5 with a step of 1.

b. For each particle in the swarm, the position vector is updated by Equation 7:

$$\vec{x_i}^{k+1} = \vec{x_i}^{k} + \vec{v_i}^{k+1}\Delta t$$

Equation 7

c. For each particle in the swarm, the position vector is reevaluated by the objective function and the local best position is set to the current position if it is better than the current local best position, as described in Equation 8:

$$\vec{p_i}^{k+1} = \begin{cases} \vec{x_i}^{k} & if\ \vec{x_i}^{k}\ is\ better\ than\ \vec{p_i}^{k} \\ \vec{p_i}^{k} & if\ \vec{p_i}^{k}\ is\ better\ than\ \vec{x_i}^{k} \end{cases}$$

Equation 8

d. The global best position of the swarm vector is set to the best among the local best positions of the particles in the swarm as described in Equation 9:

$$\vec{g}^{k+1} = \max_{\mathrm{i}}(\vec{p_i}^{k+1})$$

Equation 9

6. The final solution to the optimization problem is the detection architecture provided by the global best position $\vec{g}$ at convergence of the PSO algorithm

The PSO algorithm implemented in this CBP mission study is assumed to converge whenever the maximum change in the global best fitness is smaller than 10^{-25} for a pre-defined number of successive moves *IteNb*. However, in order to avoid stagnation of the PSO, a maximum number of generations of 300 is specified. If this number is reached, then the PSO is stopped and other parameter settings must be determined so that the PSO can converge to a solution.

Finally, the population size, the swarm confidence factor and the second convergence criterion need to be investigated further as described in subsequent sections.

6.1.6. Concluding Remarks on the Selection and Modification of Optimization Algorithms to Optimize the Customs and Border Protection Scenario

The sections above have addressed the selection and the modification of evolutionary optimization algorithms such as genetic algorithm (GA) and particle swarm optimization (PSO), to solve the multi-objective, discontinuous and non-linear optimization problem considered in this research. The aforementioned optimization algorithms have been modified in order to:

- more efficiently balance the tradeoff between exploration and exploitation,
- adequately find a number of optimally reliable solutions rather than a single solution for the distributed system architectures in specific operational contexts,
- carefully handle performance and/or cost constraints,
- thoroughly explore the search space with smaller numbers of objective function evaluations.

These sections addressed the optimization method research question, and validated the corresponding hypothesis.

6.2. Testing of the Modified Genetic Algorithm and Particle Swarm Optimization Algorithm

Evolutionary optimization algorithms such as GA or PSO have been shown to present convergence issues for highly dimensional, discontinuous, non-linear problems, due to the dependence of the algorithm parameters on the nature of the problem to which they are applied. The efficiency of a GA or PSO algorithm highly depends on the operators, on the parameters settings, and on the particular convergence criterion. Consequently, it is important to determine a set of parameters adapted to the CBP optimization problem to ensure good convergence properties and realistic solutions.

In fact, unsuccessful preliminary runs of the modified GA and of the modified PSO algorithm on the CBP optimization problem were performed using values of parameters reported in similar sensor placement studies published in the literature [160],[169],[142],[146],[170],[171],[172],[168],[161],[162],[164],[155],[165],[156],[173],[166]. These runs revealed that both algorithms would not converge. In this preliminary study, the mutation rate was varied from 0.15% to 15%, the stopping criterion for convergence of the GA was set to thirty generations, the particle swarm confidence factor was set to two, the stopping criterion for convergence of the PSO was set to two thousand generations, and finally the population size was set to a hundred architectures. Besides, the maximum number of systems allowed in the detection architecture *S_Max* was set to fifty to mitigate the dimensionality of the problem for this first trial of the modified evolutionary algorithms. In this context, it is both meaningful and critical to investigate why the modified GA and the modified PSO algorithm would not converge to a solution when using the aforementioned settings for the optimization parameters.

6.2.1. Selection of Testing Functions

In order to determine the most pertinent set of algorithm parameters for the CBP optimization problem, the modified GA and PSO algorithm are first applied to simpler analytical test problems (to which the analytical solutions are known) presenting similar discontinuous, non-linear, and multi-dimensional properties as the original problem. The algorithm parameters specific to the GA and to the PSO algorithm may be varied so as to provide a way to analyze the sensitivity of the resulting test solutions to their combination and their settings. Eventually, the set of parameter values that provides the most accurate solution for the test problems can be determined. Then, one can assume that using the resulting set of GA and PSO algorithm parameter values on the original CBP optimization problem may ensure the convergence of the optimization algorithms to adequate solutions.

A set of testing functions has been identified from the literature on genetic algorithms, evolutionary strategies, and global optimization [465],[466],[467] to evaluate and compare the performance of the modified GA and of the modified PSO algorithm as a function of their main optimization parameters. The test functions considered are the first De Jong's function [468], the Schwefel's function [469], the Rastrigin's function [470], the Griewangk's function [471], the Ackley Path Function [472], and the Michalewicz's function [473]. All of the above test functions are scalable, meaning that they can be applied to as many dimensions as necessary through the adjustment of a single parameter value inside the function.

6.2.1.1. First De Jong's Function

The first De Jong's function, also known as the sphere model, is continuous, convex, and unimodal. It is defined in Equation 10:

$$f_{De\_Jong}(x) = \sum_{i=1}^{n} x_i^2 \quad \text{for} -5.12 \leq x_i \leq 5.12$$

Equation 10

The first De Jong's function has a global minimum located at $x^* = 0$ such that $f(x^*) = 0$, as depicted in Figure 49 for three variables x_1, x_2, and x_3.

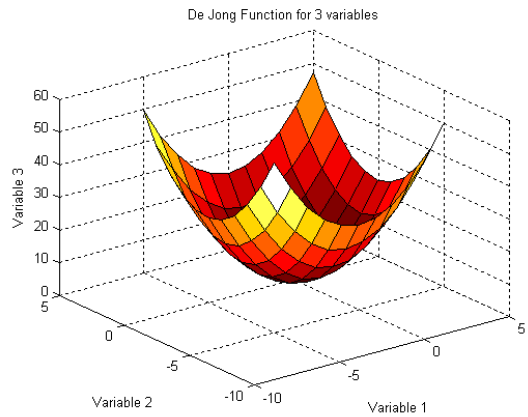


Figure 49: Graphical Representation of the First De Jong's Test Function

6.2.1.2. Schwefel's Function

The Schwefel's function is characterized by a parameter space in which the global minimum is geometrically distant from the next best local minima. In this case, the optimization algorithm may tend to mistakenly converge to a local minimum. The Schwefel's function is defined in Equation 11:

$$f_{Schwefel}(x) = \sum_{i=1}^{n} -x_i sin\left(\sqrt{|x_i|}\right) \text{ for } -500 \leq x_i \leq 500$$

Equation 11

The Schwefel's function has a global minimum located at $x^* = 420.9687$ such that $f(x^*) = -n*418.9829$, as depicted in Figure 50 for three variables x_1, x_2, and x_3.

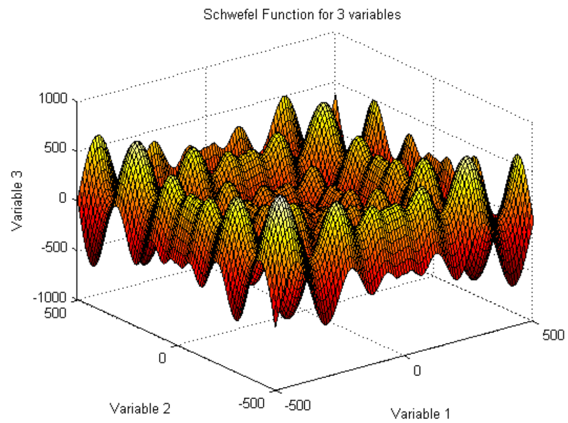


Figure 50: Graphical Representation of the Schwefel's Test Function

6.2.1.3. Rastrigin's Function

The Rastrigin's function is an extension of the first De Jong's function. It adds a cosine modulation to the pure sum of squares to generate a multitude of regularly distributed local minima. The Rastrigin's function is thus highly multimodal. It is defined in Equation 12:

$$f_{Rastrigin}(x) = \sum_{i=1}^{n} (\quad {}_i^2 - 10 * \cos\left(2\pi x_i\right) \quad \text{for} - 5.12 \leq x_i \leq 5.12$$

Equation 12

The Rastrigin's function has a global minimum located at $x^* = 0$ such that $f(x^*) = 0$, as depicted in Figure 51 for three variables x_1, x_2, and x_3.

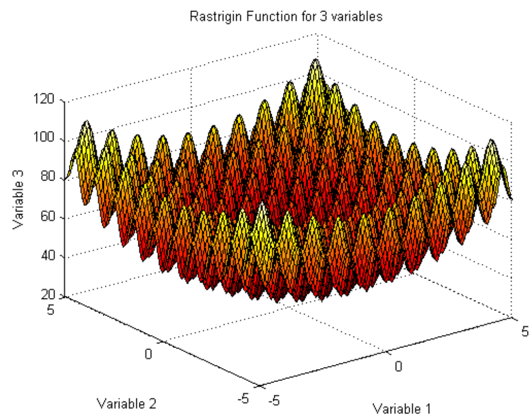


Figure 51: Graphical Representation of the Rastrigin's Test Function

6.2.1.4. Griewangk's Function

The Griewangk's function is similar to the Rastrigin's function with the exception that the local minima are widely spread over the parameter space. Their location is nevertheless regularly distributed. The Griewangk's function is defined in Equation 13:

$$f_{Griewangk}(x) = \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \text{ for} - 600 \leq x_i \leq 600$$

Equation 13

The Griewangk's function has a global minimum located at $x^* = 0$ such that $f(x^*) = 0$, as depicted in Figure 52 for three variables x_1, x_2, and x_3 and three different ranges of the parameter space. One can notice that on the full range of definition, the Griewangk's function is very similar to the first De Jong's function. However, as one zooms in on the inner area, several small peaks and valleys appear and become smooth near the global optimum value.

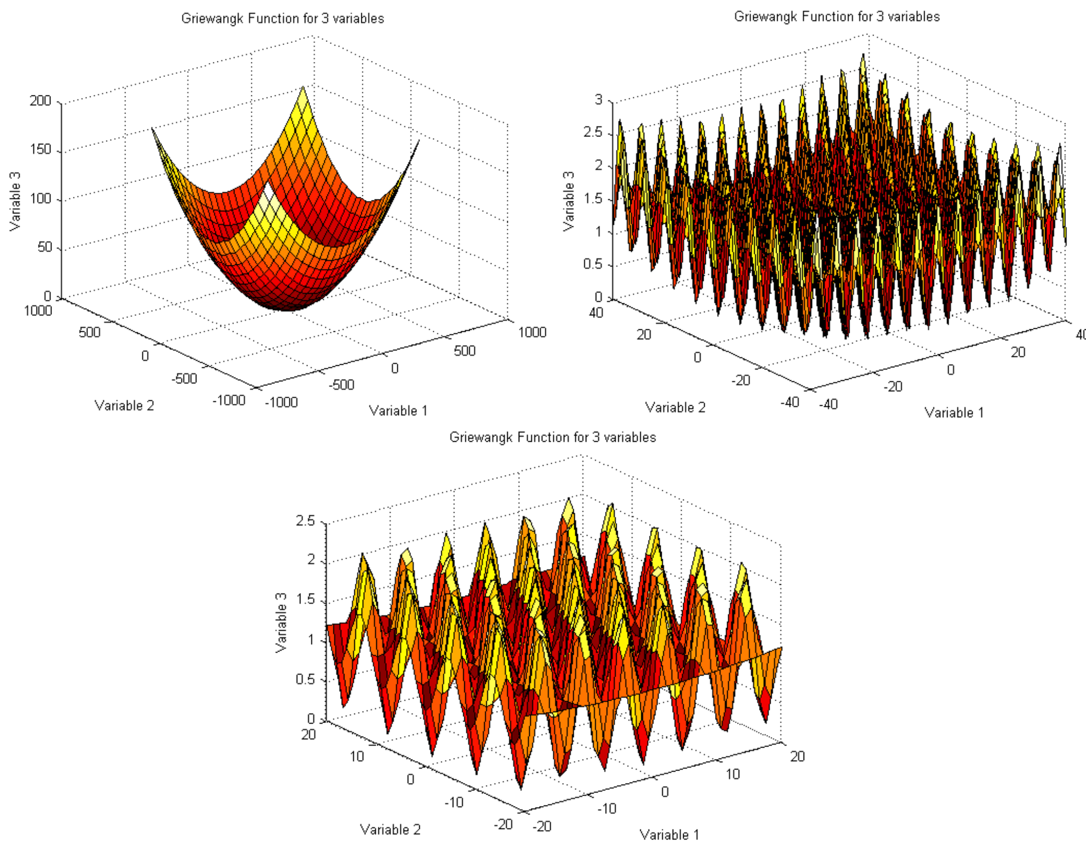


Figure 52: Graphical Representation of the Griewangk's Test Function at Increasing Zoom-Levels

6.2.1.5. Ackley Path Function

The Ackley Path function is a multimodal function defined in Equation 14:

$$f_{Ackley}(x) = -a * e^{-b\sqrt{\frac{\sum_{i=1}^{n} x_i^2}{n}}} - e^{\frac{\sum_{i=1}^{n} \cos(cx_i)}{n}} + a + e^1 \text{ for} - 1 \leq x_i \leq 1$$

Equation 14

For a = 20, b = 0.2, and c = 2π, $-32.768 \leq x_i \leq 32.768$.

The Ackley Path function has a global minimum located at *x** = 0 such that f(*x**) = 0, as depicted in Figure 53 for three variables x_1, x_2, and x_3, and the above values for a, b, and c.

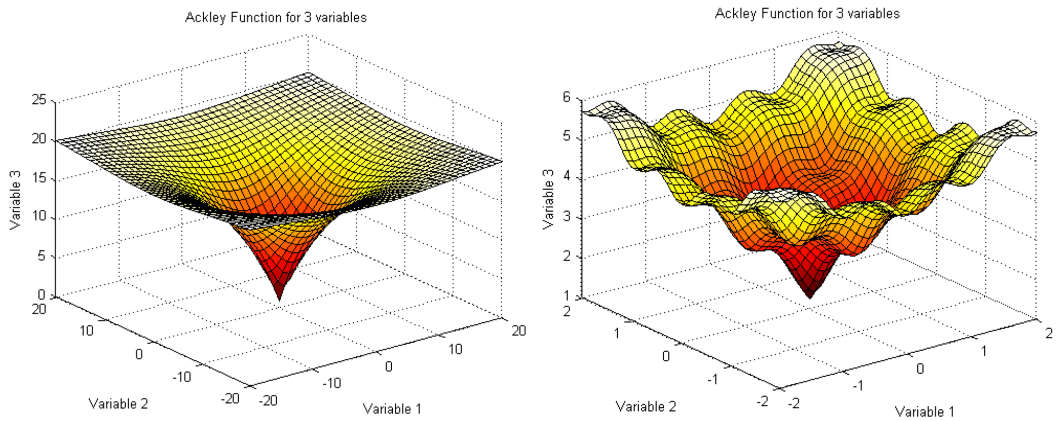


Figure 53: Graphical Representation of the Ackley Path Test Function at Increasing Zoom-Levels

6.2.1.6. Michalewicz's Function

The Michalewicz's function is a multimodal function composed of n-factorial local optima and defined in Equation 15:

$$f_{Michalewicz}(x) = -\sum_{i=1}^{n} \sin(x_i) * \left(sin\left(\frac{i * x_i^2}{\pi} \right) \right)^{2m} \quad \text{for } 0 \leq x_i \leq \pi$$

Equation 15

The Michalewicz's function is depicted in Figure 54 for three variables x_1, x_2, and x_3, and m = 10. In the Michalewicz's function, m represents the "steepness" of the valleys such that larger m values lead to steeper valleys or needles and thus harder search for the global optimum, while smaller m values lead to flatter valleys or plateaus and thus easier search. The Michalewicz's function is depicted in Figure 55: Graphical Representation of the Michalewicz's Test Function for Different Values of the Parameter m for three variables x_1, x_2, and x_3, and for different values of the parameter m = {1; 5; 10; 50; 100; 150}.

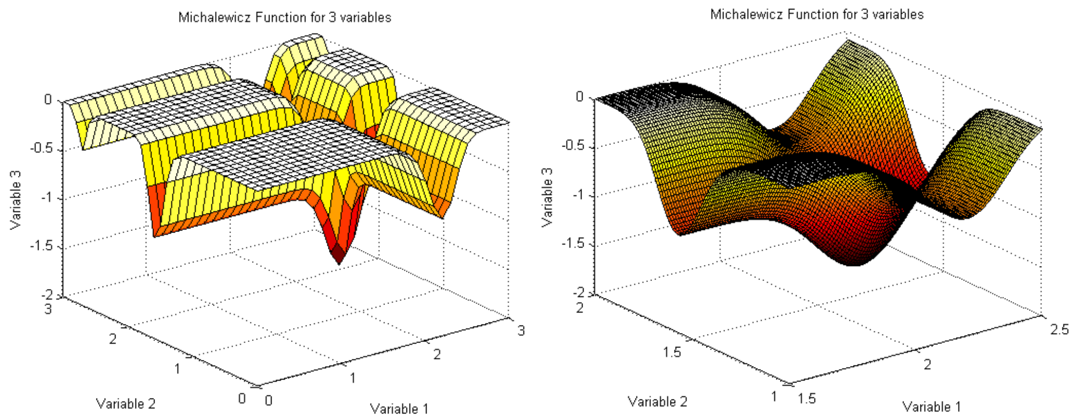
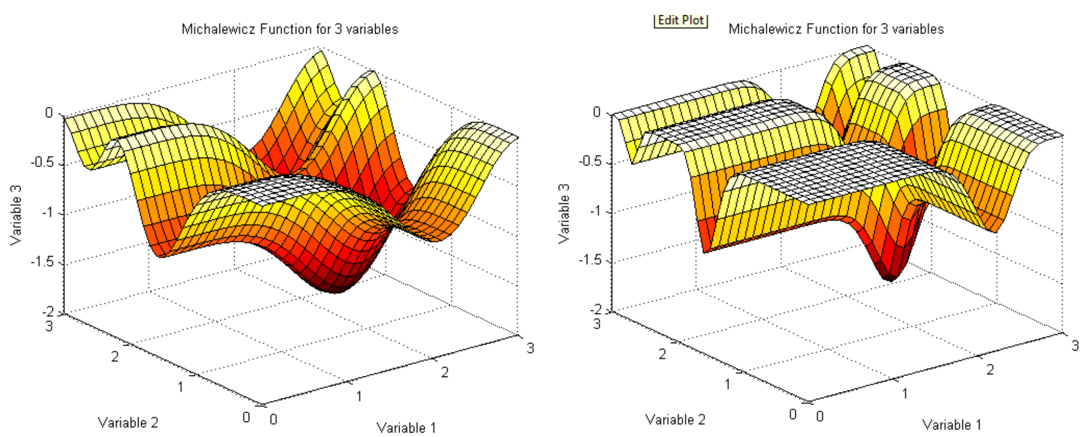
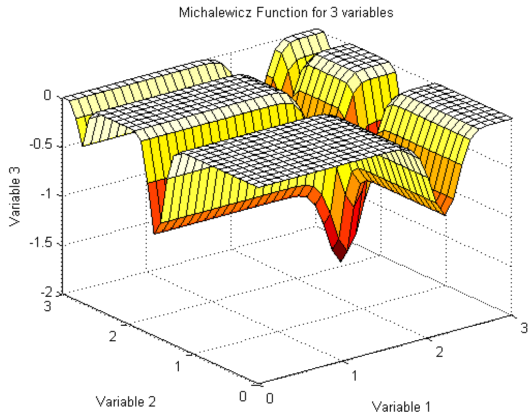


Figure 54: Graphical Representation of the Michalewicz's Test Function

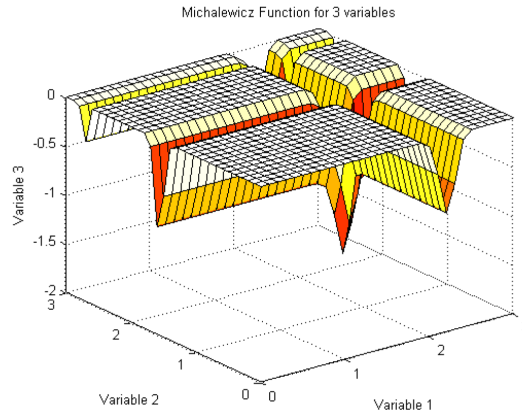


m* = *1 ***m* = *5***

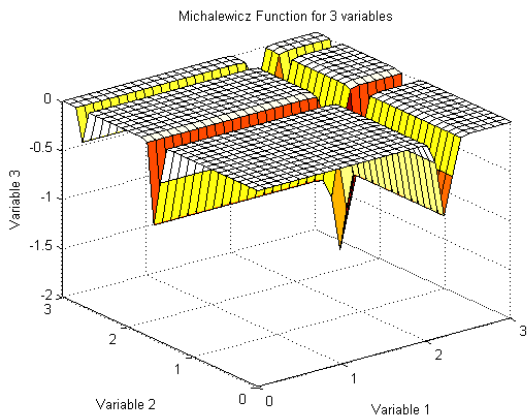
Figure 55: Graphical Representation of the Michalewicz's Test Function for Different Values of the Parameter *m*



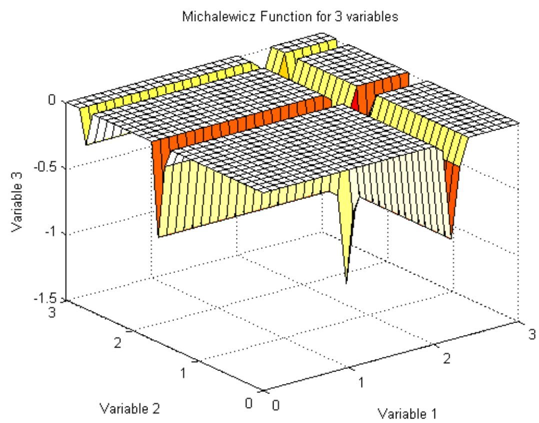
m = 10



m = 30



m = 50



m = 100

Figure 55: Graphical Representation of the Michalewicz's Test Function for Different Values of the Parameter *m* (Continued)

6.2.1.7. Selection of a Reduced Set of Test Functions

Several empirical and experimental studies have attempted to compare particular optimization algorithms and to show their effectiveness at solving a wide variety of problems. Nevertheless, attempting to analyze the performance of optimization algorithms on empirical test functions may be dangerous in that they are typically designed to be robust and general purpose search tools. One obvious drawback to such an approach is that the resulting solutions to the empirical evaluation of optimization algorithms depend as much on the testing problems as on the algorithms themselves. This can lead to specialized algorithms tuned to perform well on particular test functions but not necessarily adapted to the original optimization problem to be solved. Therefore, it is of interest to choose a suite of test functions that are not only challenging and diverse as per their design space, but also that present similar properties as the original optimization problem.

In the case of the CBP mission scenario, the optimization problem is mixed, non-smooth, highly dimensional, and has a large and multimodal search space. All of the test functions examined in the previous sections can be applied to as many dimensions as required and can be modified to feature discrete variables by slicing the corresponding design space where necessary. Therefore, the only additional properties they need to have to perform a relevant evaluation and comparison of optimization algorithms are their multimodality and their ability to handle both continuous and discontinuous variables. Table 15 summarizes the selection of the reduced set of test functions.

Table 15: Comparison and Selection of Relevant Test Functions for the Modified GA and the Modified PSO Algorithm

Test Function / *Criteria*	**First De Jong**	**Schwefel**	**Rastrigin**	**Griewangk**	**Ackley**	**Michalewicz**
Challenging	✗	✓	✓	✓	✗	✗
Unusual	✗	✓	✓	✗	✗	✗
Mixed	✗	✓	✓	✓	✓	✓
Highly Dimensional	✓	✓	✓	✓	✓	✓
Multimodal	✓	✓	✓	✓	✓	✓
Large Search Space	✓	✓	✓	✓	✓	✓

Legend:

- ✓ satisfies property
- ✗ does not satisfy property

As can be noticed from Table 15, the **Schwefel**'s function and the **Rastrigin**'s function turn out to be the most appropriate functions to evaluate and compare the performance of the modified GA and of the modified PSO algorithm at solving the CBP optimization problem.

The Rastrigin's function is highly multimodal, composed of many local optima of identical shapes and depths, regularly distributed, and encompassed in a bowl-shaped design space, at the center of which the global optimum lies.

The Schwefel's function is also highly multimodal, composed of a multitude of local optima spread over a relatively larger design space, and more distant from the global optimum. The optima also have different shapes and depths. This makes it harder for the evolutionary algorithms to find the true optimum of the Schwefel's function lying at the bottom of a steeper bowl-shaped envelop.

In both cases, the optimization algorithms may tend to mistakenly converge to a local optimum and that is why the Rastrigin's function and the Schwefel's function have been identified as challenging and of interest for the analysis of the optimization parameters required to solve the CBP optimization problem.

6.2.2. Tuning of the Optimization Algorithm Parameter Settings for the Modified Genetic Algorithm and Particle Swarm Optimization Algorithm

The reduced set of test functions selected in the previous section can now be used to investigate the optimum parameterization of optimization operators for both the GA and the PSO algorithm. For the GA, the algorithm parameters of interest encompass the population size, the mutation rate, and the number of successive generations over which the objective function has not improved significantly. For the PSO algorithm, they are the population size, the swarm confidence factor, and the number of successive moves over which the objective function value has not changed significantly. In addition to the above parameters, the maximum number of detection systems allowed in the architecture *S_Max* plays a significant role in the optimization process and is worth investigating as well. Indeed, this parameter defines the dimensionality of the optimization problem in that it gives the "length" of the chromosomes and of the particles in the GA and the PSO respectively. As a reminder, the length of each individual in a population is *n*S_Max*, where n is the number of variables in the problem ($n = 4$ for the CBP mission scenario).

Therefore, the modified GA and the modified PSO algorithm can now be applied to the Schwefel's function and the Rastrigin's function under various combinations of their respective optimization parameters in order to analyze the sensitivity of the resulting solutions to the values of the aforementioned parameters. The goal is thus to determine the appropriate combination of the population size (*PopSize*), the mutation rate (*MutRate*), the number of successive generations where no improvement in the objective function can be observed (*StopIte*), and *S_Max* for the modified GA, and of the population size (*PopSize*), the swarm confidence factor (C_2), the number of successive moves over which the objective function value has not changed significantly (*IteNb*), and *S_Max* for the modified PSO. Table 16 summarizes the ranges of values of the optimization parameters investigated in this sensitivity study.

Table 16: Ranges of Values for the Optimization Parameters in the Sensitivity Study on the Reduced Set of Test Functions

Parameter	Minimum Value	Maximum Value	Step
Population Size *PopSize*	50	200	30
Mutation Rate *MutRate*	0.05	0.5	0.05
StopIte	30	110	20
Swarm Confidence Factor C_2	2	2.5	0.1
IteNb	1600	2000	100
S_Max	10	190	20

If one were to consider all the possible combinations of the above optimization parameters, one would need to run 3000 (6*10*5*10) and 1800 (6*6*5*10) sensitivity cases for the modified GA and for the modified PSO respectively. In order to speed up the analysis, Designs of Experiment (DoEs) are used.

6.2.2.1. Designs of Experiment

Designs of Experiment were created in the 1920's for agricultural applications. Since then, they have been widely used in the industrial and systems engineering fields. An Experimental Design is actually the laying out of a detailed experimental plan in advance of conducting an experiment. Well chosen experimental designs maximize the amount of "information" that can be obtained for a given amount of experimental effort.

Designs of Experiment are composed of a series of tests in which purposeful changes are made to the input variables so that one may observe and identify the reasons for change in an output response.

The advantage of using DoE is that a maximum amount of knowledge can be gained with a minimum expenditure of experimental effort. According to the Engineering Statistics Handbook [474], a Design of Experiment is "*a systematic, rigorous approach to engineering problem solving that applies principles and techniques at the data collection stage so as to ensure the generation of valid, defensible and supportable engineering conclusions. In addition, all this is carried under the constraint of a minimal expenditure of engineering runs, time and money*."

There exist four general engineering problem areas in which DoE may be used:

1. Comparative
2. Screening/Characterizing
3. Modeling
4. Optimizing

In a comparative design, the engineer is interested in assessing whether a change in a single factor has resulted in a change/improvement of the process as a whole.

In a screening/characterizing design, the engineer is interested in "understanding" the process as a whole in the sense that he/she wishes (after design and analysis) to have in hand a ranked list of the most significant to the least significant factors that affect the process.

In a modeling design, the engineer is interested in functionally modeling the process with the output being a good fitting mathematical function (high predictive power), and to have good estimates of the coefficients in that function (maximal accuracy).

In an optimizing design, the engineer is interested in determining optimal settings of the process factors, i.e. to determine for each factor, the level of that factor which optimizes the process response.

For instance, the sensitivity analysis of the performance of the modified GA and modified PSO algorithm to their parametric settings using test functions is a combination of optimizing design and screening design. First, the decision maker is interested in determining combinations of optimization operator values that enable the modified algorithms to efficiently converge to the true global optima of the Schwefel's function and Rastrigin's function. In this case, the resulting optimal parameter settings are assumed to be optimized for the characteristics of the original problem. Second, the decision maker is interested in performing a screening test on the various optimization operators and their settings so as to study the sensitivity of the algorithm performance to the parameter values. in this case, the goal is to understand the optimization process as a whole and to have, in hand, a ranked list of the most significant to the least significant optimization factors affecting the convergence of the algorithms.

In this respect, DoEs are typically used for several purposes, namely:

- Choosing between alternatives: for instance, choosing between different combinations of the optimization parameters for each tested algorithm
- Selecting the key factors affecting the response of interest: for instance, selecting the key optimization factors affecting the convergence of the optimization algorithms to the test solutions
- Performing response surface modeling

- Hitting a target: for example, converging to the global optimum of optimization problems of the same type as the test functions
- Reducing variability: for example, reducing variability in the performance of the optimization algorithms with respect to the optimization problem considered
- Maximizing or minimizing a response: for example, maximizing the ability of the optimization algorithms to converge to the solutions of problems presenting similar characteristics as the test functions
- Making a process robust, i.e. insensitive to noise variables that are beyond the control of the designer: for example, making the optimization algorithms robust to various optimization formulations for a given category of problems
- Seeking multiple goals: for example, finding the global optimum of the test functions while optimizing the combination of optimization operator values to obtain the best performance of the optimization algorithms when solving the initial CBP mission problem

- Performing a regression modeling

Furthermore, the choice of an initial DoE is problem-dependent and is based on the number of independent variables in the model, the speed (or execution time) of the analysis tool(s), the overall accuracy desired, and the behavior of the response.

Typically, six types of designs are used by engineers:

1. Full Factorial Designs
2. Latin Hypercube Designs
3. Box-Behnken Designs
4. Central Composite Designs, especially Face Centered Central Composite Designs
5. D-Optimal Designs
6. Custom Designs

Figure 56 provides a visual representation of the first four classical designs in three dimensions.

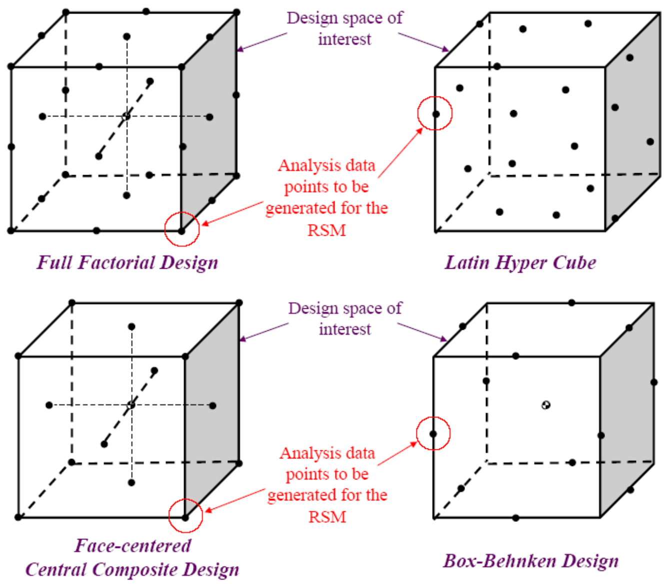


Figure 56: Visual Representation of Four of the Five Classical Designs Used in Engineering

From Figure 56, one can see that the overlay of a Box-Behnken design with a three-level Face Centered Central Composite Design results in a complete three-level full factorial design.

Table 17 summarizes the advantages and disadvantages of the designs depicted in Figure 56, and gives the corresponding number of runs in the DoE (n is the total number of design variables).

Table 17: Advantages and Disadvantages of Four of the Five Classical Designs Used in Engineering

Design	Advantages	Disadvantages	Number of Runs
Full Factorial	1. Every point is considered, which reduces error 2. Orthogonal design	1. Excessively high number of cases to test 2. Limited to second order or less functions	(Number of levels)^n
Latin Hypercube	1. Rich sampling of interior of design space 2. Highly accurate on interior 3. Greater than second order polynomial functions can be considered	1. Possible high correlation of independent variables 2. Poor accuracy on edges of design space	User Specified
Central Composite	1. Extremes of the design space considered 2. Extrapolation minimized 3. Orthogonal design	1. Large design space can result in many unconverged solutions 2. Limited to second order functions	2^n+2n+1
Box-Behnken	1. Better convergence of analysis tools 2. Fewer executions required 3. Orthogonal design	1. Extrapolation to extremes of design space introduces error for non-linear design spaces 2. Maximum 16 variables 3. Limited to second order functions	No equation available (3 variables require 15 runs, 7 variables require 57 runs)

A D-optimal design is an n-factor, second order design where a minimum number of points, namely $\frac{(n+1)(n+2)}{2}$ (n being the total number of design variables) is used to minimize the variance of the data by maximizing the determinant of |X'X|, X being the design matrix of variables involved in the DoE. A D-optimal design is a saturated design for two reasons:

1. It uses the minimum number of points possible, i.e. the number of experiments is equal to the number of points considered.
2. It is characterized by the fact that main effects are not confounded with each other, but main effects can be confounded with two-factor interactions.

Due to the second property mentioned above, a D-optimal design is not orthogonal since it does not guarantee that the effects of one factor or interaction on the response of interest can be estimated independently of any other factor or interaction. In other words, it is not guaranteed that the effects of one factor or interaction on the response of interest, is clear of any influence due to any other factor or interaction.

Finally, a Custom Design is a design where the user personally defines the various levels of the design variables and the discretization of the design space based on a-priori knowledge about the problem.

To conclude, DoEs are characterized by the richness of the sample space (i.e. the data points to be evaluated), the correlation of independent variables (i.e. the closeness of the design to an orthogonal design), and the number of data points required to yield a good representation of the system under study. For a more detailed description of the fundamentals and above properties of Designs of Experiment, the reader may refer to [475] and [476], among other references.

In order to study the sensitivity of the performance of the modified GA and of the modified PSO to settings of their respective optimization parameters using the Schwefel's function and the Rastrigin's function, a Space Filling Design (Latin Hypercube Design) and a Custom Design have been created. Both designs are based on the ranges of values identified in Table 16 for each of the optimization parameters considered. The resulting Space Filling Designs are provided in Appendix J for both the modified GA and the modified PSO, while the Custom Designs are provided in Appendix K for both the modified GA and the modified PSO respectively.

6.2.2.2. Optimized Parameter Settings for the Modified Genetic Algorithm

6.2.2.2.1. Test Functions Analysis – Sensitivity of the Modified Genetic Algorithm to the Optimization Parameters

Figure 57 graphs the sensitivity of the converged solutions for the *Schwefel's function* and the *Rastrigin's function* to the mutation rate, the number of successive generations where no improvement in the objective function can be observed, and the population size. This sensitivity analysis is performed for both the Custom DoE and the Space Filling DoE and for different ranges of values of the maximum number of systems allowed in the architecture *S_Max*. In this study, the Schwefel's function is scaled so that the global optimum value is $f(x^*) = 0$. In Figure 57, each black dot corresponds to a DoE case. For each setting or each range of values of *S_Max* considered, the blue horizontal line further represents the fit to the fitness values in the dimension of *MutRate*, the red horizontal line models the fit to the fitness values in the dimension of *StopIte*, and the green horizontal line corresponds to the fit to the fitness values in the dimension of *PopSize*.

Figure 57 shows that larger values of *S_Max* result in a smaller sensitivity of the solution to the optimization parameter settings and in a reduced spread of the solution values around the optimum of 0. Indeed, as *S_Max* increases from 10 to 200, the spread in the solutions decreases significantly, and the mean of the converged solution approaches the true optimum of 0 for *S_Max* as low as 60.

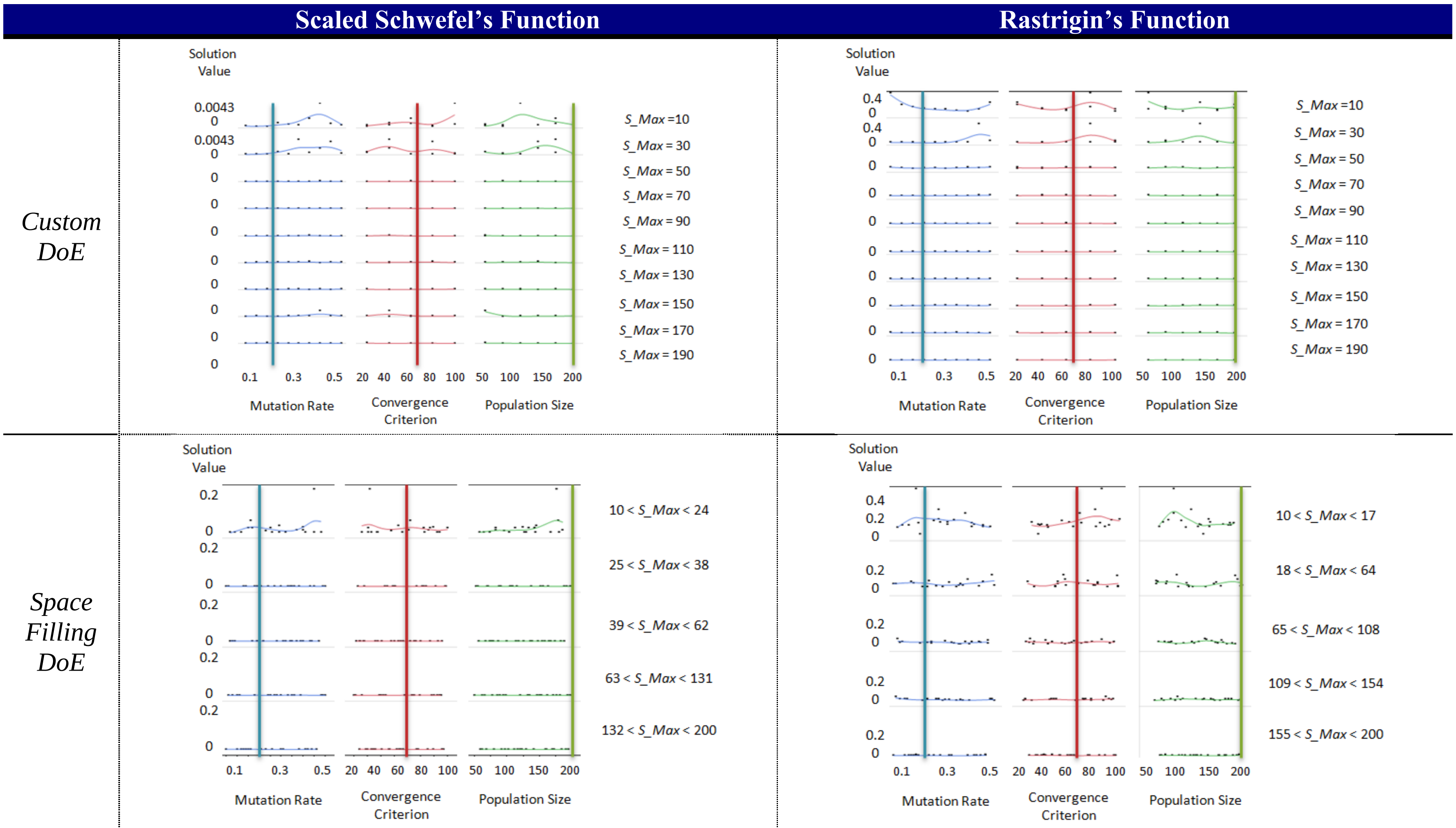


Figure 57: Sensitivity of the Convergence of the Modified GA for Different Values of the Optimization Parameters and Different Settings of *S_Max*

6.2.2.2.2. Test Functions Analysis – Mean Value and Spread in the Solutions

The distribution function of the solutions for each bucket of *S_Max* is investigated and does not exhibit any specific shape, which was expected. In order to quantify the spread in the solutions, the standard deviation statistics is used. The results are summarized and depicted in Figure 58. Larger values of *S_Max* lead to a more consistent convergence of the modified GA which manages to find the true optimum of the test functions for *S_Max* as low as 70 in the case of the custom DoE. In the case of the Space Filling DoE, the modified GA manages to find the true optimum test solutions for *S_Max* as low as 40 for the Schwefel's function and about 110 for the Rastrigin's function.

As a consequence, the predominant factor for the convergence of the modified GA is the value of *S_Max* which modulates the sensitivity of the algorithm performance to the combination of its optimization parameters. For sufficiently large values of *S_Max*, superior to 110, any combination of optimization parameter values may be considered. This is depicted in Figure 57 by the blue, red, and green vertical bars pointing at the values of the mutation rate, the convergence criterion, and the population size that have been selected as the tuned parameter settings for the modified GA. For values of *S_Max* smaller than 40, the convergence of the GA is dependent on the optimization parameter values. Therefore, the tuned parameter settings have been chosen to ensure the convergence of the modified GA to test solutions close to the true optimum values of zero for *S_Max* smaller than 40.

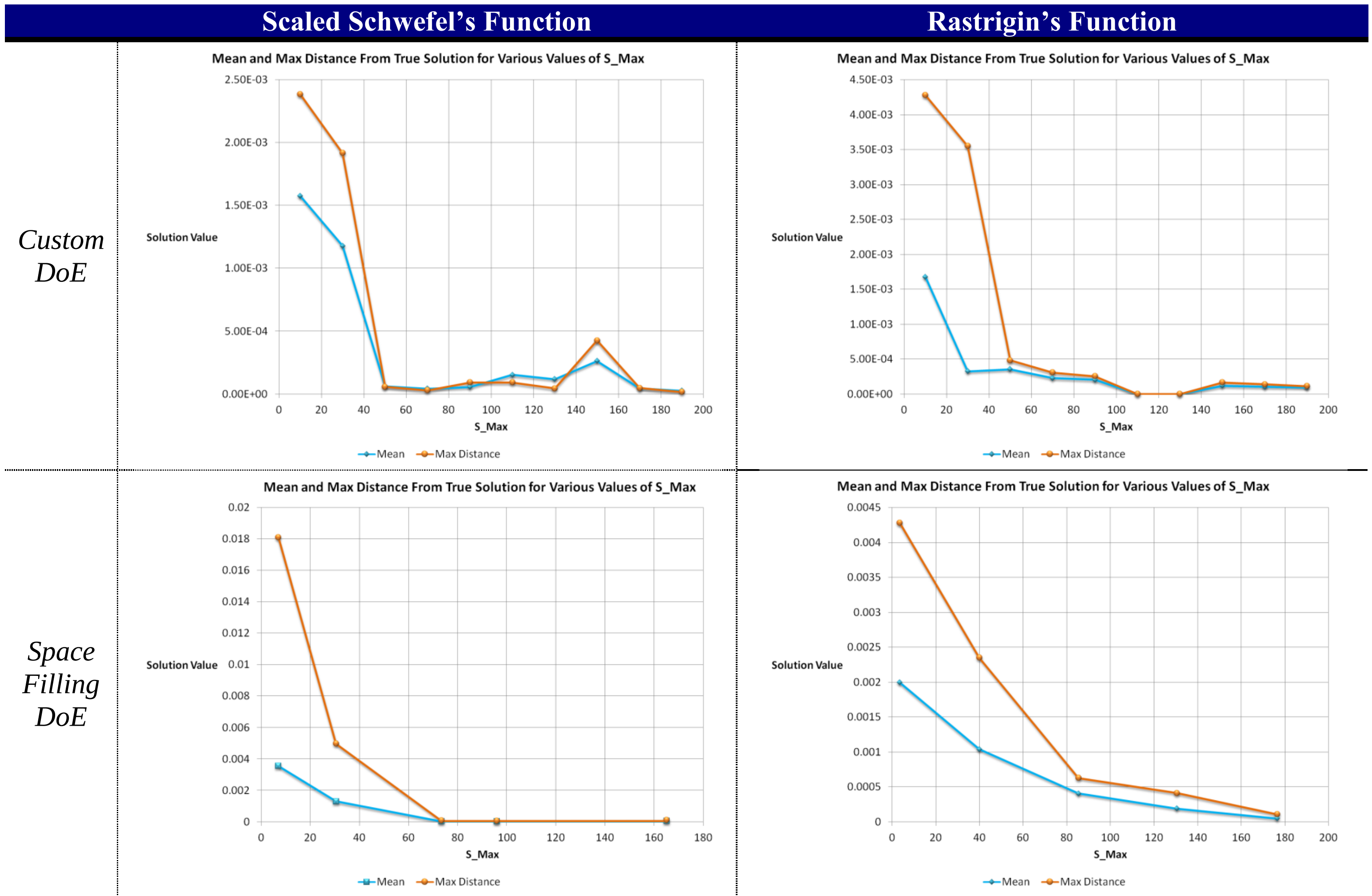


Figure 58: Plot of the Means and of the Maximum Distances Fom the True Solutions for Different Settings of *S_Max*, in the Case of the Modified GA

6.2.2.2.3. Genetic Algorithm – Tuned Optimization Parameter Settings

From Figure 57 and Figure 58 depicting the convergence characteristics of the modified genetic algorithm when applied to both the scaled Schwefel's function and the Rastrigin's function, one can conclude that larger values of the maximum number of systems allowed in the detection architecture (*S_Max*) lead to a smaller sensitivity of the algorithm to the optimization parameter values. Furthermore, as *S_Max* increases, the standard deviation in the solution space decreases. In other words, the convergence of the modified GA to a solution for the test functions is enhanced as *S_Max* increases, regardless of the settings of the optimization parameters. As a consequence, the predominant factor for the convergence of the modified GA is the value of *S_Max* which modulates the sensitivity of the algorithm performance to the combination of its optimization parameters. For sufficiently large values of *S_Max*, superior to 110, any combination of optimization parameter values may be considered. This may be explained by the fact that larger populations of chromosomes enhance the exploration capability of the modified GA and make it better able to find a solution to the optimization problem.

For the rest of the study, the following values shall be used:

- Population size = 200
- Mutation rate = 0.2 (20%)
- Number of successive generations where no improvement in the objective function can be observed = 70
- *S_Max* = 200

6.2.2.3. Optimized Parameter Settings for the Modified Particle Swarm Algorithm

6.2.2.3.1. Test Functions Analysis – Sensitivity of the Modified Genetic Algorithm to the Optimization Parameters

Figure 59 graphs the sensitivity of the converged solutions for the *Schwefel's function* and the *Rastrigin's function* to the swarm confidence factor, the number of successive moves over which the objective function value has not changed significantly, and the population size. This sensitivity analysis is performed for both the Custom DoE and the Space Filling DoE and for different ranges of values of the maximum number of systems allowed in the architecture *S_Max*. In this study, the Schwefel's function is scaled so that the global optimum value is $f(x^*) = 0$. In Figure 59, each black dot corresponds to a DoE case. For each setting or each range of values of *S_Max* considered, the blue line represents the fit to the fitness value in the dimension of C_2, the red line models the fit to the fitness value in the dimension of *IteNb*, and the green line corresponds to the fit to the fitness value in the dimension of *PopSize*.

Figure 59 shows that larger values of *S_Max* result in a smaller sensitivity of the solution to the optimization parameter settings and in a reduced spread of the solution values around the optimum of 0. Indeed, as *S_Max* increases from 10 to 200, the spread in the solutions decreases significantly, and the mean of the converged solution approaches the true optimum of 0 for *S_Max* as low as 90.

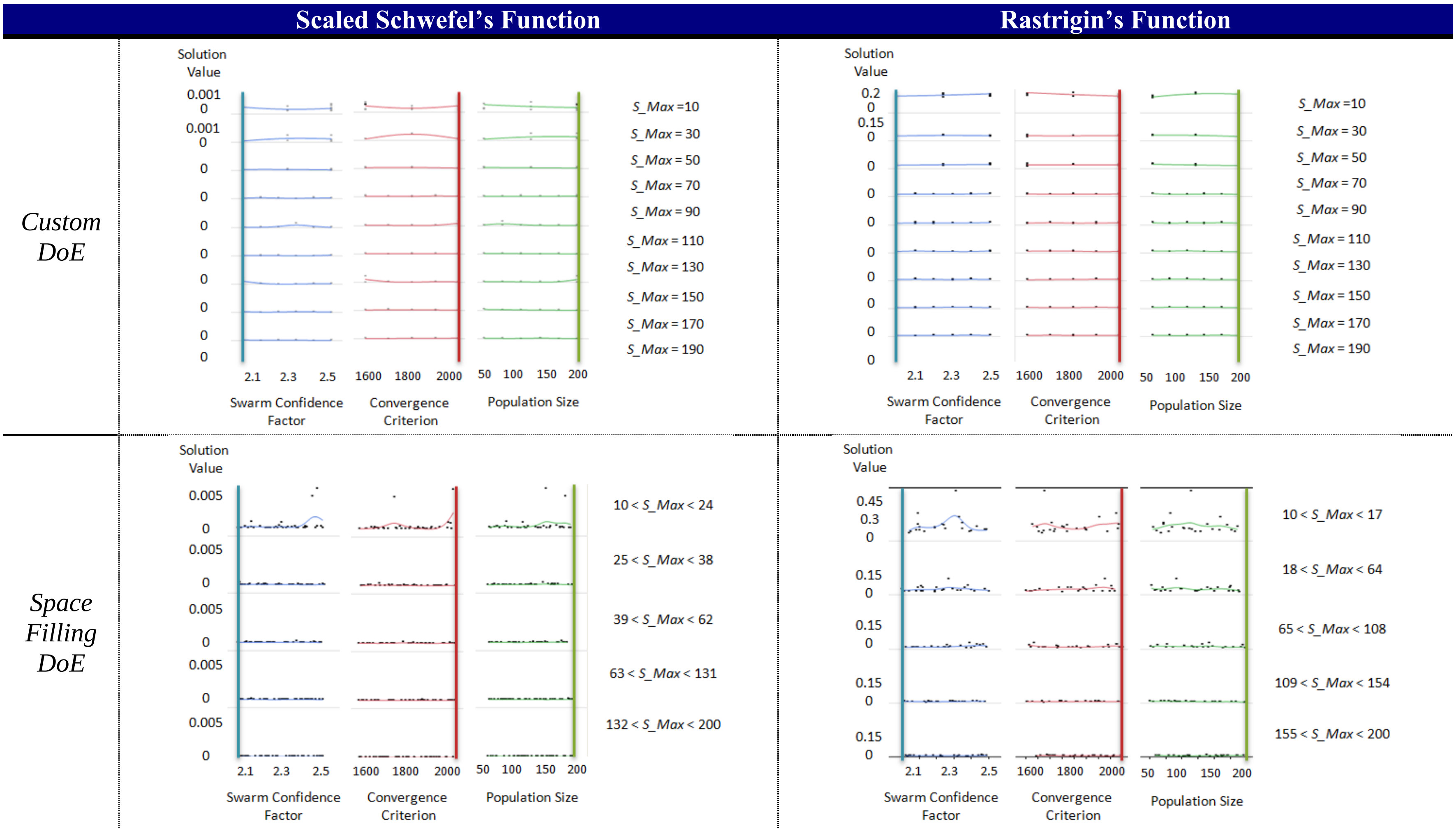


Figure 59: Sensitivity of the Convergence of the Modified PSO for Different Values of the Optimization Parameters and Different Settings of *S_Max*

6.2.2.3.2. Test Functions Analysis – Mean Value and Spread in the Solutions

The distribution function of the solutions for each bucket of *S_Max* is investigated and does not exhibit any specific shape, which was expected. In order to quantify the spread in the solutions, the standard deviation statistics is used. The results are summarized and depicted in **Figure 60**. Larger values of *S_Max* lead to a more consistent convergence of the modified PSO algorithm which manages to find the true optimum of the test functions for *S_Max* as low as 150 in the case of the Custom DoE. In the case of the Space Filling DoE, the modified PSO converges to the true optimum test solutions for *S_Max* as low as 30 for the Schwefel's function and about 110 for the Rastrigin's function.

As a consequence, the predominant factor for the convergence of the PSO algorithm is the value of *S_Max* which modulates the sensitivity of the algorithm performance to the combination of its optimization parameters. For sufficiently large values of *S_Max*, greater than 150 for the scaled Schwefel's function and greater than 70 for the Rastrigin's function, any combination of optimization parameter values may be considered. This is depicted in Figure 59 by the blue, red, and green vertical bars pointing at the values of the swarm confidence factor, the convergence criterion, and the population size that has been selected as the tuned parameter settings for the modified PSO. Similarly to what was observed for the modified GA, the convergence of the PSO algorithm is dependent on the optimization parameter values for *S_Max* smaller than 40. Therefore, the tuned parameter settings have been chosen to ensure the convergence of the modified PSO algorithm to test solutions close to the true optimum values of zero for *S_Max* smaller than 40.

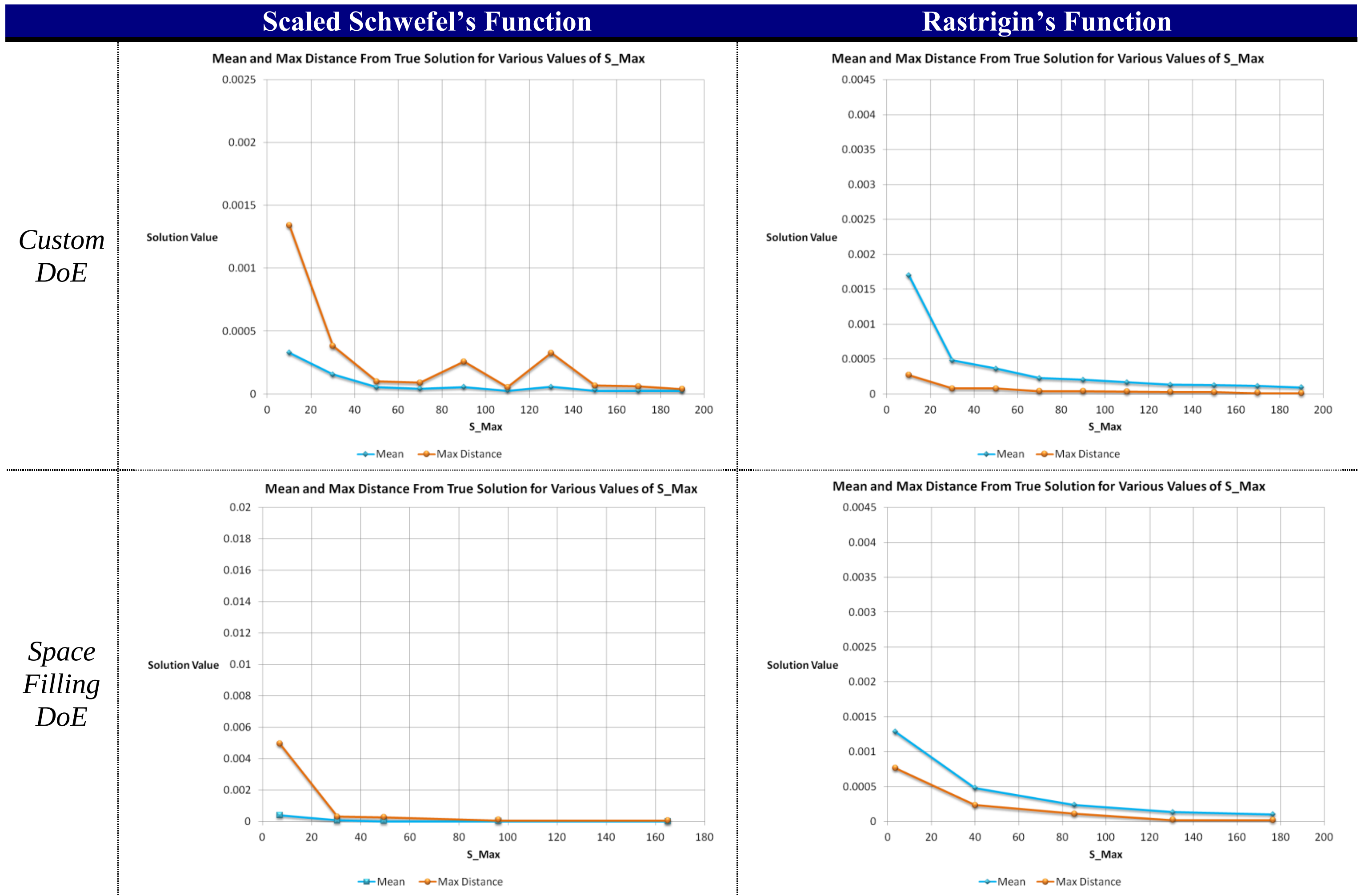


Figure 60: Plot of the Means and of the Maximum Distances From the True Solutions for Different Settings of *S_Max*, in the Case of the Modified PSO

6.2.2.3.3. Particle Swarm Optimization – Tuned Algorithm Parameter Settings

From Figure 59 and **Figure 60** depicting the convergence characteristics of the modified particle swarm optimization algorithm applied to both the scaled Schwefel's function and the Rastrigin's function, one may conclude that larger values of the maximum number of systems allowed in the detection architecture (*S_Max*) lead to a smaller sensitivity of the algorithm to the optimization parameter values. Furthermore, as *S_Max* increases, the standard deviation in the solution space decreases. In other words, the convergence of the modified PSO algorithm to a solution for the test functions is enhanced as *S_Max* increases, regardless of the settings of the optimization parameters. As a consequence, the predominant factor for the convergence of the PSO algorithm is the value of *S_Max* which modulates the sensitivity of the algorithm performance to the combination of its optimization parameters. For sufficiently large values of *S_Max*, greater than 150 for the scaled Schwefel's function and greater than 70 for the Rastrigin's function, any combination of optimization parameter values may be considered. This rather nice convergence of the modified PSO algorithm may be explained by its ability to exploit the cooperative and social aspects of evolution rather than its competitive aspects as in traditional evolutionary algorithms. The reason for the high overall performance of the modified PSO algorithm in the experiments described above may also be due to the wide and random search space involved. This implies that an increase in *S_Max* has a greater probability of reaching the global optimum at an early stage of the search. It may also be that the global nature of the search offers insight into various local neighborhoods of the search space. This is mainly due to the ability of the particles to communicate with each other and to have a knowledge and a memory of both the best location they have visited so far and the best position the whole swarm has discovered so far. It may finally be that particles moving fast towards the best position visited by the swarm allow the modified PSO to perform a detailed search of a good region of the design space at an early stage. Such features are absent or not prevalent in the modified GA.

For the rest of the study, the following values are used:

- Population size = 200
- Swarm Confidence Factor = 2
- Number of successive moves over which the objective function value has not changed significantly = 2000
- *S_Max* = 200

6.2.2.4. <u>Comparison of the Convergence Times for the Modified GA and the Modified PSO</u>

Figure 61 shows the time required for the modified GA and the modified PSO to converge to a solution to the test functions, over a series of runs corresponding to different ranges of values of *S_Ma*x as summarized in Table 18.

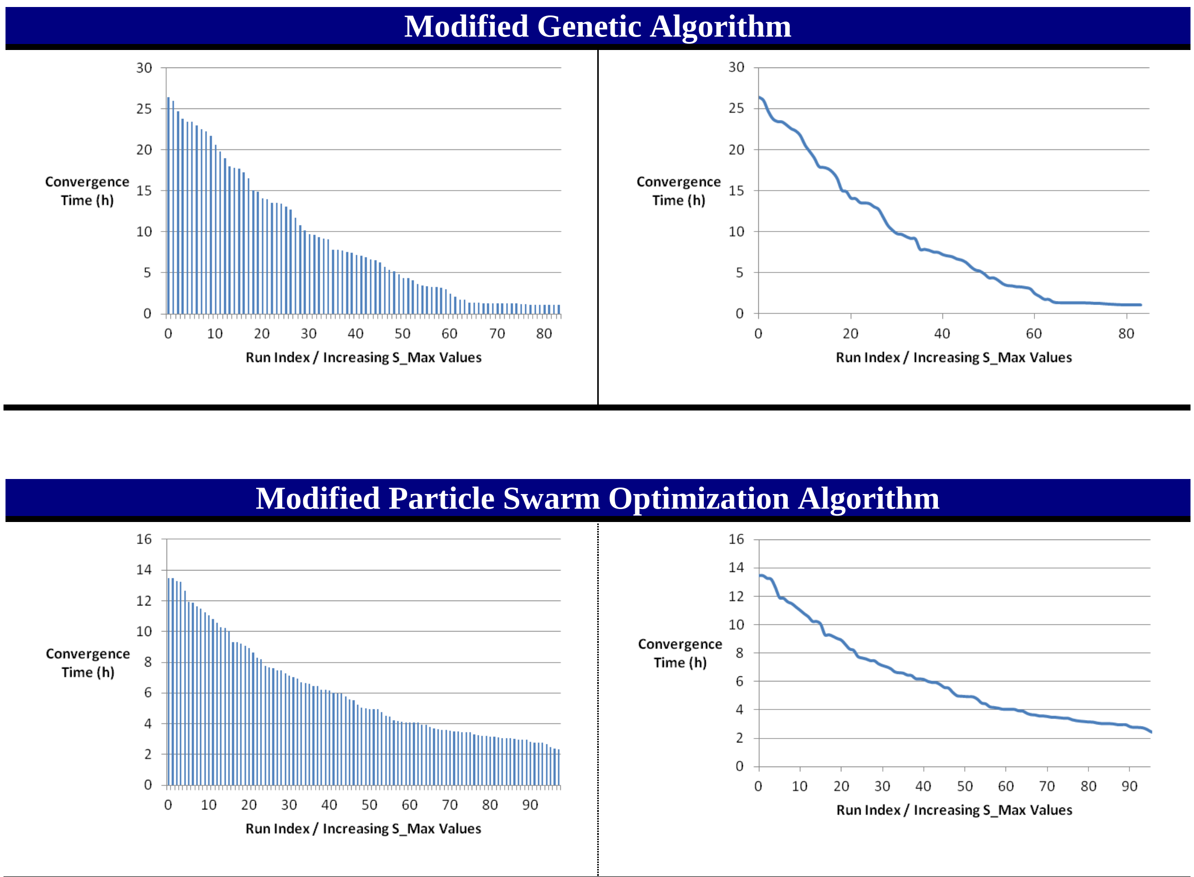


Figure 61: Time Required for the Modified GA and the Modified PSO to Converge to the Test Solutions Over Multiple Runs (Different Values of *S_Max*)

Table 18: Run Indices and Corresponding Value Ranges for *S_Max*

Modified Genetic Algorithm		Modified Particle Swarm Optimization	
DoE Runs	*S_Max Value*	*DoE Runs*	*S_Max Value*
0-8	10-30	0-10	10-30
9-17	31-50	11-21	31-50
18-26	51-70	22-32	51-70
26-34	71-90	33-43	71-90
35-43	91-110	44-54	91-110
44-52	111-130	55-65	111-130
53-65	131-150	66-76	131-150
66-74	151-170	77-87	151-170
75-83	171-190	88-99	171-190

From Figure 61, one can notice that, as *S_Max* increases, the times required for the modified GA and PSO to converge to the true solutions of the test functions decrease. This is consistent with the previous conclusions that the efficiency of the modified algorithms is enhanced by longer chromosomes and particles. This indeed translates into a more thorough exploration of the design space.

From Figure 61, one can readily notice that for values of *S_Max* smaller than 40, the PSO algorithm tends to converge faster than the GA. However, as *S_Max* increases, the time required for the GA to converge to the true solutions of the test dunctions decreases faster than that required for the PSO. The convergence times then seem to have identical evolutions for values of *S_Max* larger than 50.

The distribution function of the convergence times for each bucket of *S_Max* was investigated and did not exhibit any specific shape, which was expected. In order to draw more rigorous conclusions about which algorithm converges faster and to quantify the spread in the convergence times, the mean and the standard deviation statistics of the convergence time are calculated for both the modified GA and the modified PSO algorithm. The results of the calculations are summarized in Table 19 and Figure 62.

Table 19: Means and Standard Deviations of the Convergence Times for the Modified GA and Modified PSO Algorithm for Different Ranges of Values of *S_Max*

	GA		PSO	
Maximum Number of Systems Allowed (S_Max)	*Mean*	*Standard Deviation*	*Mean*	*Standard Deviation*
10-30	23.94	1.45	12.3	0.936
31-50	18.69	1.7	9.67	0.736
51-70	13.79	0.78	7.54	0.446
71-90	9.738	1.1	6.3	0.277
91-110	7.2	0.45	5.12	0.388
111-130	4.85	0.84	4.08	0.173
131-150	2.86	0.61	3.52	0.108
151-170	1.37	0.14	3.12	0.089
171-190	8.89	0.086	2.7	0.225

As can be derived from Figure 61, Figure 62 and Table 19, on average, the modified particle swarm optimization algorithm seems to converge much faster and in a more consistent manner for both the scaled Schwefel's function and the Rastrigin's function compared to the modified genetic algorithm. Indeed, although the modified GA converges faster on average for large values of *S_Max*, the modified PSO converges more consistently from *S_Max* ~ 100 onward. Therefore, the modified PSO algorithm seems more computationally efficient at solving optimization problems like the Schwefel's function and the Rastrigin's function presenting a mixed, discontinuous, highly dimensional, large and multimodal search space.

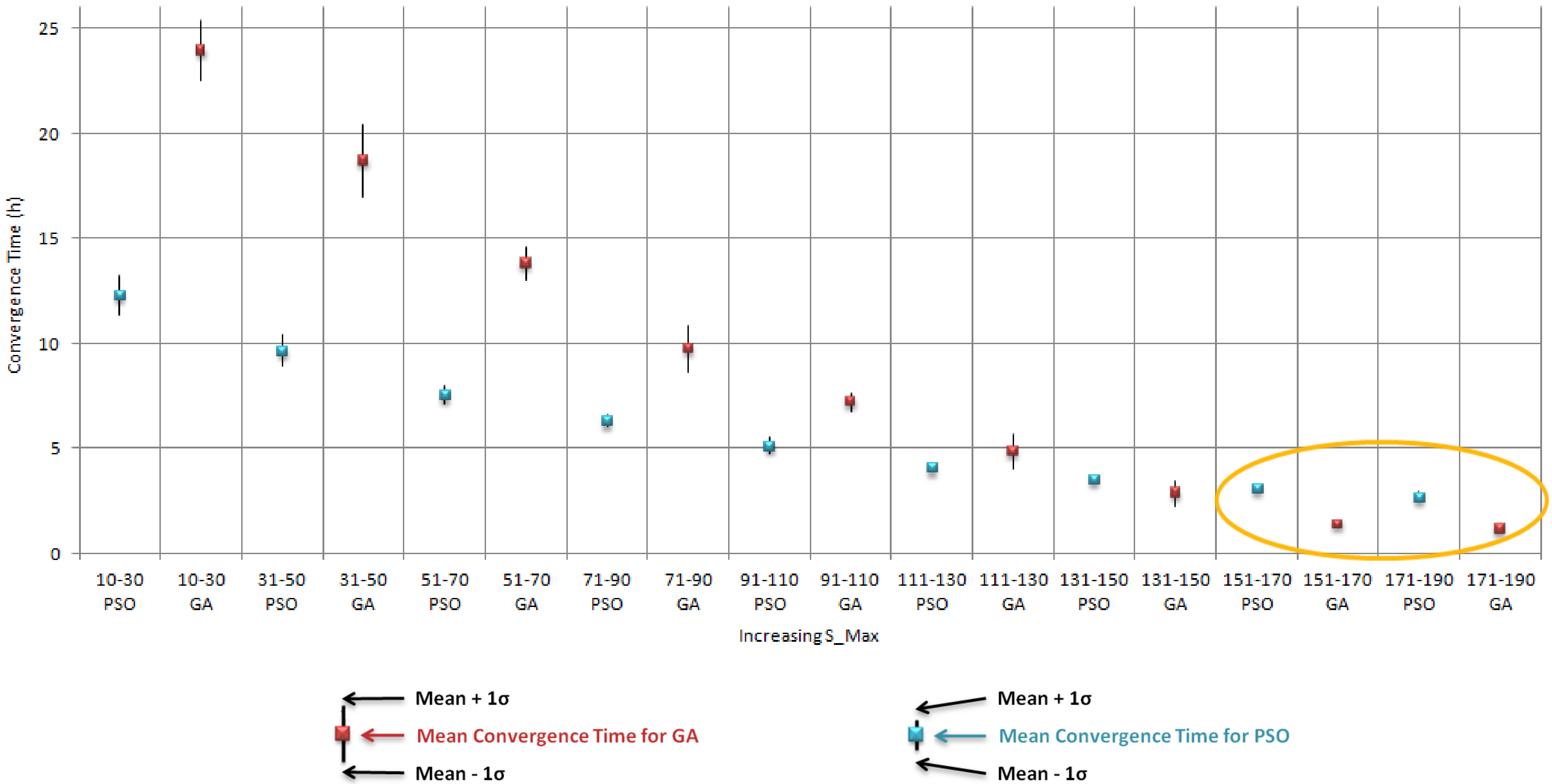


Figure 62: Means and One Sigma Boundaries of the Convergence Times for the Modified GA (Red) and the Modified PSO Algorithm (Blue), for Increasing Values of *S_Max*

6.2.2.5. Final Comparison of the Modified GA and of the Modified PSO and Selection of an Optimization Approach for the Customs and Border Protection Scenario

The information provided in the previous two sections and summarized in Figure 57 to Figure 62, shows that for both the modified GA and the modified PSO algorithm, as *S_Max* increases from 10 to 200, the standard deviation in the fitness values decreases significantly, and the mean of the converged solution approaches the true optimum of both the scaled Schwefel's function and the Rastrigin's function. In other words, larger values of *S_Max* lead to a better convergence of the modified algorithms regardless of the settings of their specific optimization parameters. In addition, both evolutionary algorithms manage to find the true optimum of the test functions for *S_Max* as low as 70 or 150 depending on the case considered. Nevertheless, the standard deviations in the solutions provided by the modified PSO algorithm for both test functions are smaller than those in the solutions provided by the modified GA. Therefore, the modified PSO algorithm seems more able to deal with optimization problems that are mixed, discontinuous, highly dimensional, and that have a large and multimodal search space. This most probably comes from the fact that PSO is based on the principle that each individual in the swarm can benefit from the discoveries and previous experiences of all the other companions during the search for the optimum. In the PSO algorithm, each particle in the swarm population is assumed to "fly" over the search space to find promising regions of the landscape. Unlike GA, PSO does not use evolutionary operators such as crossover and mutation to manipulate individuals of the swarm. Rather, each individual in the swarm flies in the search space with a "velocity" which is dynamically adjusted according to its own flying experience and to that of its companions. Hence, PSO has the ability to search effectively large spaces. Additionally, the modified PSO algorithm does not suffer from the same difficulties as the modified GA in that the progress towards the solution is enhanced and not detracted by the interaction between the particles (individuals) of the group (swarm population). A particle swarm system also has a memory: each particle keeps both the memory of its own best position and of the group's best position. In the modified PSO algorithm, individuals who fly past optima are made to return towards them since knowledge of good solutions is retained by all particles. On the contrary, changes in the genetic populations in the modified GA result in the destruction of previous knowledge of the problem, except when the best individual or the first several best individuals of a given generation is/are automatically passed to the next generation through an elitist reproduction approach. In the later case, one or a small number of individuals effectively retain their "identities" (or memories) as they are passed to the next population. This may further explain why PSO converges faster to the test solutions than GA when applied to both the scaled Schwefel's function and the Rastrigin's function as depicted in Figure 61. Indeed, it was shown that on average, the modified PSO algorithm converges one and half times faster for both test functions compared to the modified GA, and that the standard deviation in the time required for the modified PSO algorithm to converge to the test solutions was about two and a half times smaller than that for the modified GA.

To summarize, PSO has better performance, lower computational cost and is easy to implement. Due to its population-based solution mechanisms, PSO is suitable for multi-disciplinary optimization and is capable of providing several solutions in one execution. The reason for the high efficiency of the PSO algorithm in the experiments described above may be that the coverage of the search space is random and wide so that an increase in the population size and in *S_Max* has a greater probability of reaching the global optimum at an early stage of the search. In addition, the global nature of the search in the modified PSO algorithm offers insight into various local neighborhoods of the design space. Finally, particles moving fast towards the best position visited by the swarm allow the modified PSO algorithm to perform a detailed search of a good region at an early stage. Due to all these advantages of PSO over GA, only the modified PSO algorithm will be investigated subsequently and applied to the CBP optimization problem.

Finally, it is of interest to remind the reader about the results from the preliminary study where the modified GA and the modified PSO algorithm were applied to the original CBP optimization problem using algorithm parameter settings found in the literature. In this case, the maximum number of detection systems allowed in the architecture solution (*S_Max*) was set to fifty. It was shown that both the modified GA and the modified PSO algorithm could not converge. One can now conclude that this was due to *S_Max* being too small for the algorithms to be insensitive to the settings of their respective optimization parameters. In this context, the actual algorithm parameter values were not adapted for the convergence of the algorithms to a solution. This is easily seen in Figure 57 for the modified GA, where a mutation rate varying from 0.15% to 15% combined with a stopping criterion for convergence of the GA of 30 generations and a population size of 100 lead to a large standard deviation in the solutions to both the scaled Schwefel's function and the Rastrigin's function for *S_Max* = 50. Similarly, a particle swarm confidence factor of 2 combined with a stopping criterion for convergence of the PSO of 2000, and a population size of 100 correspond to a large standard deviation in the solutions to both the scaled Schwefel's function and the Rastrigin's function for *S_Max* = 50. This is easily seen in Figure 59 for the modified PSO algorithm.

6.2.3. Concluding Remarks on the Tuning of the Optimization Algorithm Parameter Settings

The previous sections have demonstrated the development of a rigorous, structured and traceable approach to determine a set of optimization algorithm parameters that ensures good convergence properties and the adequacy of the resulting solutions in the context of surveillance and protection missions for homeland security. This was done by applying the modified GA and the modified PSO algorithms to a set of simpler analytical test problems (whose solutions are known) presenting similar discontinuous, non-linear, and dimensional properties as the original homeland security application. Then, the algorithm parameters specific to each optimization approach were varied so as to analyze the sensitivity of the solutions to the optimization parameter settings and combinations, and to determine the set of algorithm parameter values that provides the most accurate solution for the test problems. Additionally, the performance of the modified evolutionary algorithms at solving the test functions was compared and the optimization method which globally presented the best performance and the lowest computational cost was selected. Finally, it was assumed that the resulting set of algorithm parameter values is able to ensure the convergence of the optimization algorithm to accurate distribution system architecture solutions for the homeland security application of interest.

These sections addressed the optimization parameter settings research question, and validated the corresponding hypothesis.

6.3. Heuristic Optimization as a Benchmark for the Evolutionary Optimization

Evolutionary optimization algorithms such as GA and PSO have been shown to yield solutions that may not always be reproducible for large dimensions, discontinuous, non-linear problems such as the CBP optimization problem studied in this research. In such a case, it is of the utmost importance to develop an internal verification loop of the proposed optimization approach in order to check the accuracy of the solutions provided by these evolutionary optimization algorithms. This may be done by developing a heuristic recursive optimization scheme based on simple performance, cost and geometrical positioning rules, and applying it to the original CBP optimization problem. The outcome of the heuristic approach will then serve as a benchmark for the detection architectures provided by the modified GA or the modified PSO algorithm. The heuristic approach is based on the following:

- The detection systems are those defined in Table 12. They are represented by an index corresponding to their type: 1 for the HCR, 2 for the MCR, 3 for the LCR, 4 for the HCC, 5 for the MCC, and 6 for the LCC.
- The maximum number of systems allowed in the final detection architecture *S_Max* is set to 200.
- A minimum coverage *Min_Cov* and/or a maximum cost *Max_Cost* for the final detection architecture may be specified (for instance minimum coverage of 70% and maximum cost of 50 M$).
- The detection systems are preferentially located at most promising positions provided in Appendix I and depicted in the NetLogo environment in Figure 41.
- Each type of system located at a particular promising position is associated with a relative coverage efficiency defined in Equation 2. The architecture coverage is then calculated using Equation 3. Appendix L summarizes the relative coverage efficiencies for each type of system when located at each of the most promising positions.

Two different types of heuristic recursive optimization approaches have been developed. The first one generates a detection architecture composed of a single type of sensor system, while the second one generates a more realistic detection architecture composed of various types of detection systems. Both heuristic recursive optimization algorithms involve the following storage variables:

- *Sensor_list*: stores the indices associated with the types of detection systems in the current detection architecture. It is composed of integers between 1 and 6.
- *Occupied_pos_list:* stores the indices of the currently occupied promising positions (positions at which systems in the current detection architecture are located).
- *Occupied_list:* stores the latitudes, longitudes, and indices of the promising positions occupied by the systems in the current detection architecture.
- *Available_pos:* stores the latitudes, longitudes, and indices of the promising positions not currently occupied by the detection systems in the detection architecture. It corresponds to the complete table of promising positions provided in Appendix I minus the *Occupied_list*.
- *Candidate_pos:* gives the position index at which the new detection system ought to be located. It is determined using the NetLogo environment. A pre-determined detection system is located at each of the currently available promising positions and the coverage of the resulting detection architecture is calculated. Then, the detection architecture with the highest coverage provides the next candidate promising location at which the detection system will be placed.
- *Current_Architecture_Coverage:* gives the coverage of the current detection architecture.
- *Current_Architecture_Cost:* gives the cost of the current detection architecture.

6.3.1. Single Type Systems Recursive Optimization

The single type detection systems recursive optimization approach is composed of the following steps:

1. For each type of detection system considered in this study (HCR, MCR, LCR, LCC, MCC, and HCC):
    a. The *Sensor_list* variable is initialized to the index between 1 and 6 corresponding to the type of the only detection system that will be considered in the optimization (obtained from step 1).
    b. The *Available_pos* variable is initialized to the complete list of most promising positions provided in Appendix I.
    c. Considering all the promising positions in the *Available_pos* variable, the one yielding the highest coverage effectiveness value for the detection system considered in step 1 is obtained from Appendix L.
    d. The cost of the current detection architecture *Current_Architecture_Cost* is initialized to the cost of the detection system considered in step 1.
    e. The *Occupied_pos_list* variable is initialized to the index of the promising position yielding the highest coverage effectiveness value obtained in step *c* for the detection system considered in step 1.
    f. The coverage of the current detection architecture *Current_Architecture_Coverage* is initialized to the coverage effectiveness value obtained in step *c* for the detection system considered in step 1.
    g. The *Occupied_pos* variable is initialized to the latitude, longitude, and index of the promising position obtained in step *c* for the detection system considered in step 1.
    h. The *Available_pos* variable is updated based on the *Occupied_pos* variable obtained in step *g*.
    i. While the number of sensor systems in the current detection architecture is less than *S_Max* and/or the coverage of the current

detection architecture is less than *Min_Cov* and/or the cost of the current detection architecture is not more than *Max_Cost*, a new detection system of the same type as the one considered in step 1 is added to the current detection architecture based on the following operations:

ii. For the type of detection system considered, located at each available promising position stored in the *Available_pos* variable, the coverage of the resulting detection architecture is calculated using the NetLogo environment. This enables taking into account the potential overlap in coverage of detection systems in the architecture.

iii. The resulting detection architecture presenting the highest coverage value amongst all the candidate detection architectures obtained previously provides the new detection architecture. The promising position at which the new detection system will be located immediately results.

iv. The *Sensor_list* variable is updated with the type of the only detection system involved in the optimization and considered in step 1.

v. The cost of the current detection architecture *Current_Architecture_Cost* is updated with the cost of the detection system considered in step 1.

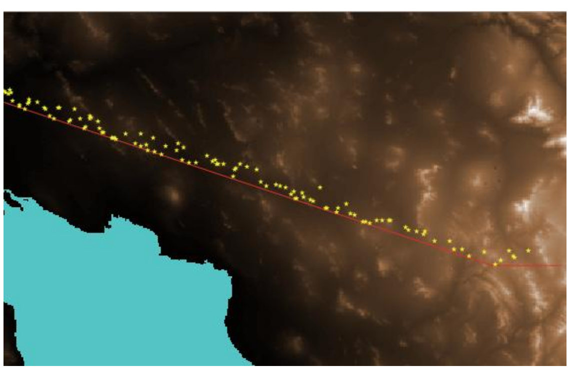
vi. The *Occupied_pos_list* variable is updated with the index of the promising position obtained in step *iii*, at which the new detection system is located.

vii. The coverage of the current detection architecture *Current_Architecture_Coverage* is updated to the coverage of the newly generated detection architecture determined from steps *ii* and *iii*.

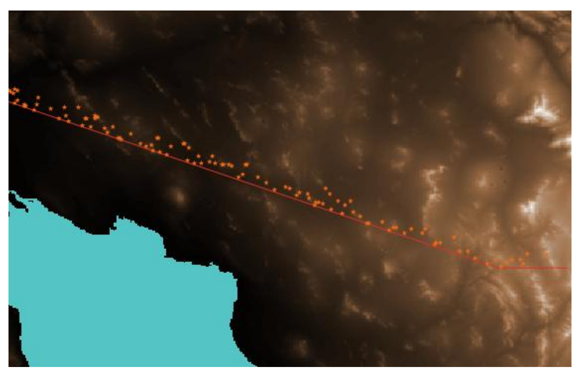
viii. The *Occupied_pos* variable is updated with the latitude, longitude, and index of the promising position obtained in step *iii*, at which the new detection system is located.

ix. The *Available_pos* variable is updated based on the *Occupied_pos* variable obtained in step *viii*.

The algorithm described above is applied to each type of detection system (HCR, MCR, LCR, HCC, MCC, and LCC) independently. The results of the heuristic recursive optimization approach considering only one type of detection system to generate a uniform detection architecture presenting the maximum coverage at the minimum cost given a constraint on *S_Max* are provided in Figure 63 for each type of detection system considered. In this study, a maximum on the total number of systems in the architecture *S_Max* of 200 and a minimum coverage value *Min_Cov* of 90% were specified. Finally, the constraint on the maximum cost of the final detection architecture *Max_Cost* was not specified. That is why, the total number of systems in the resulting detection architectures may be less than *S_Max* provided that the coverage of the said architectures is more than *Min_Cov*.

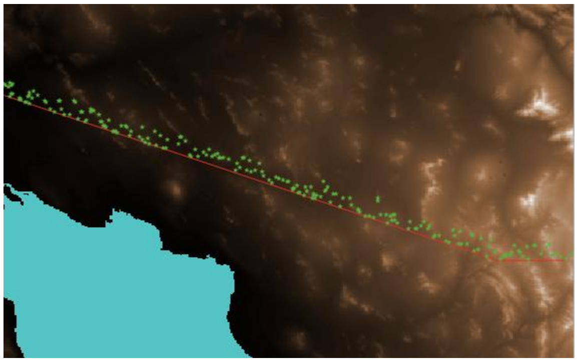


High Cost Radar Detection Architecture: 109 systems

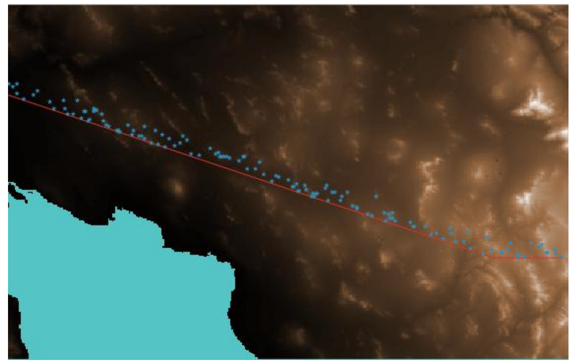


Medium Cost Radar Detection Architecture: 109 systems

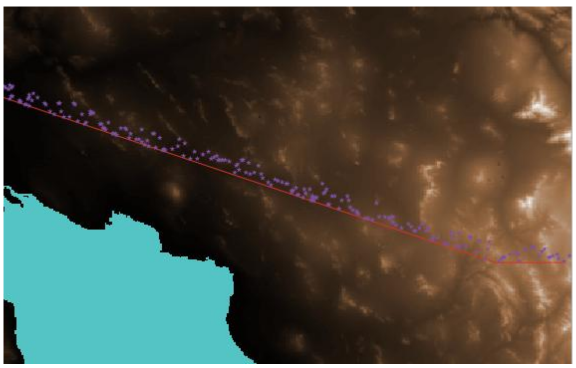
Figure 63: Optimized Detection Architectures Provided by the Recursive Optimization Approach for Each Type of Detection System



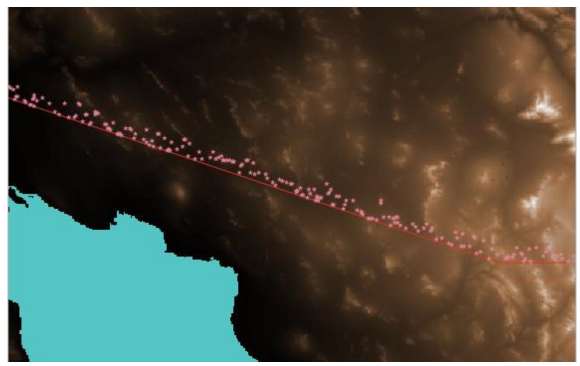
Low Cost Radar Detection Architecture:
200 systems



High Cost Camera Detection Architecture:
133 systems



Medium Cost Camera Detection Architecture:
200 systems



Low Cost Camera Detection Architecture:
200 systems

Figure 63: Optimized Detection Architectures Provided by the Recursive Optimization Approach for Each Type of Detection System (Continued)

As may be observed in Figure 63, the optimized detection architectures have very similar structures on the terrain independently of the type of detection system considered. The detection systems tend to be preferentially located at the same best positions, where they have the highest coverage efficiency.

Table 20, Figure 64, Figure 65 and Figure 66 summarize the coverage and the cost of the resulting detection architectures for each type of sensor system considered.

Table 20: Summary of the Coverage and Cost of the Single Type Systems Detection Architectures

System Type	Coverage (%)	Cost (M$)
HCR	90.6	82.5
MCR	92.4	64
LCR	87.2	37.2
HCC	90.7	22.9
MCC	80.4	8.6
LCC	50.7	2.9

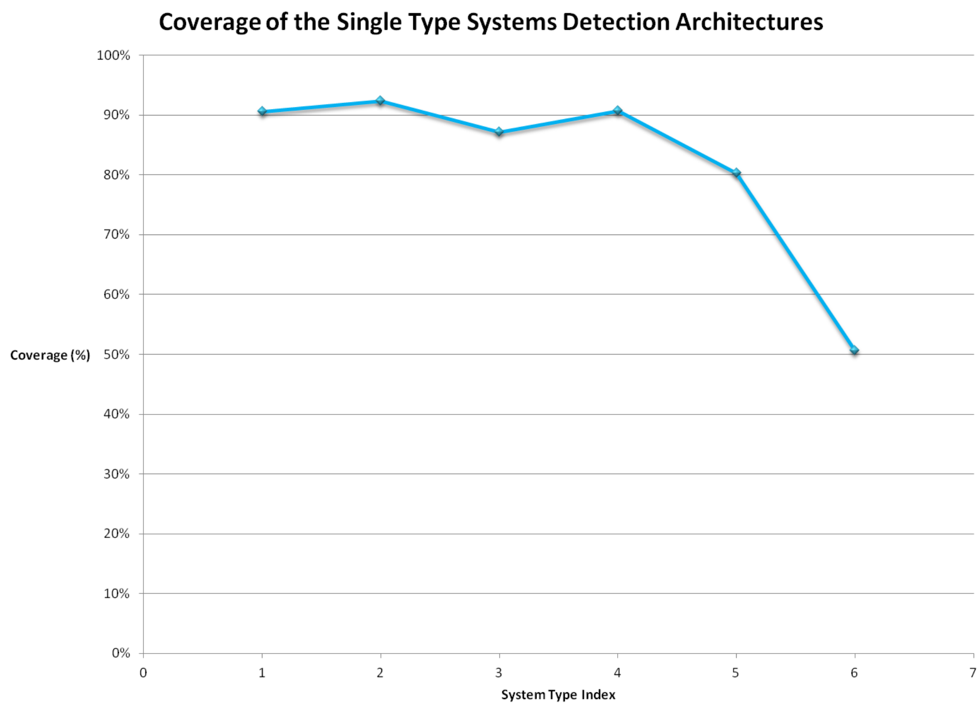


Figure 64: Summary of the Coverage of the Single Type Systems Detection Architectures

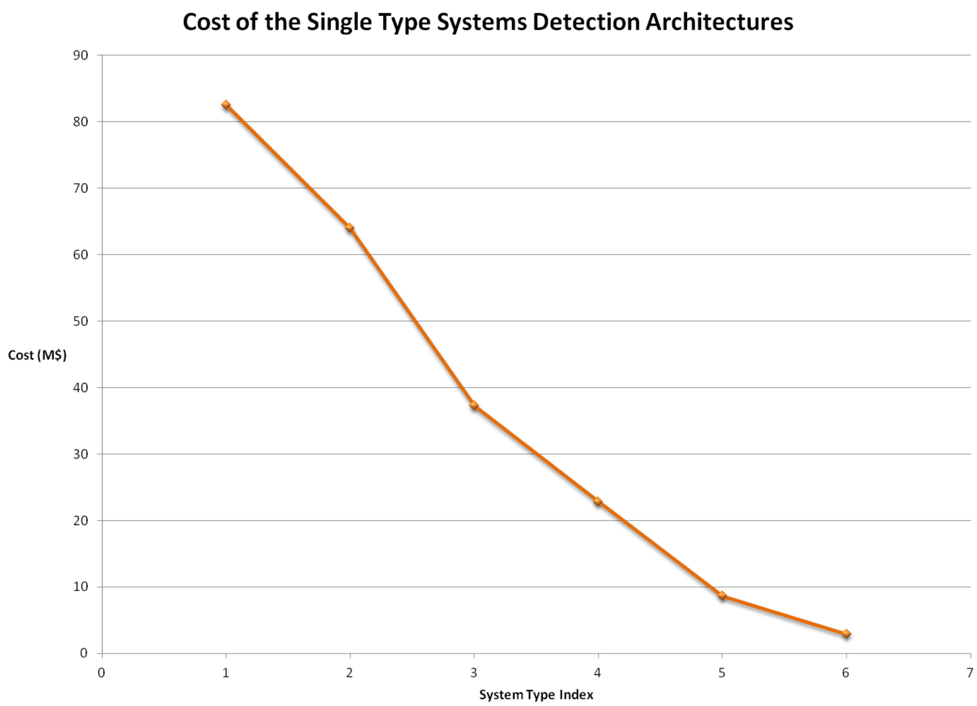


Figure 65: Summary of the Cost of the Single Type Systems Detection Architectures

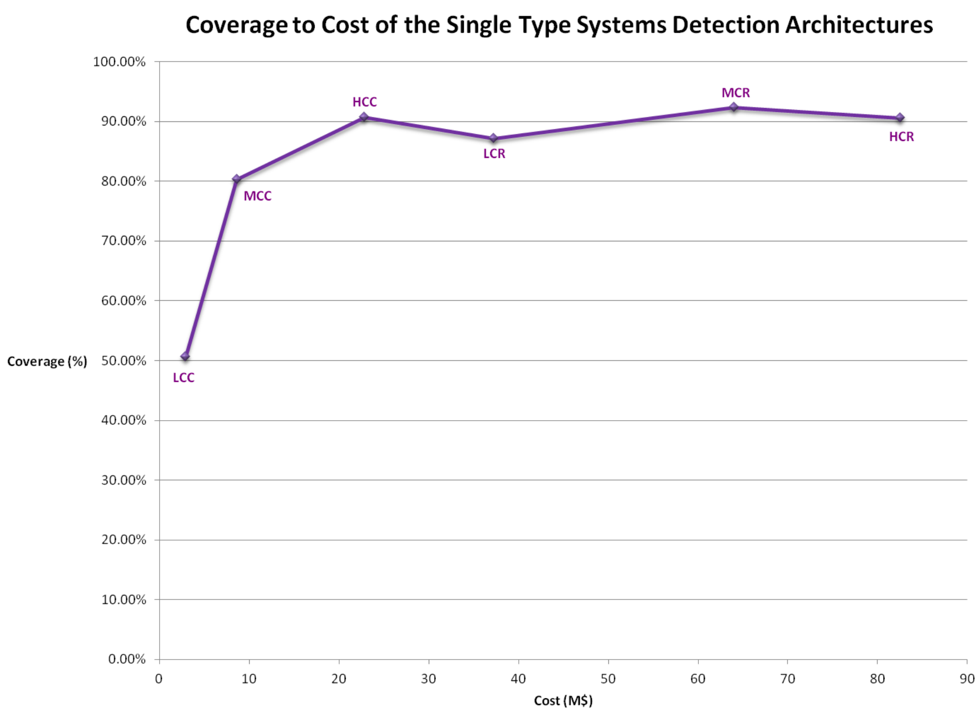
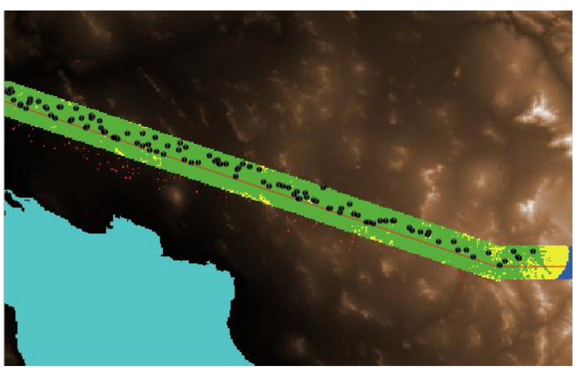


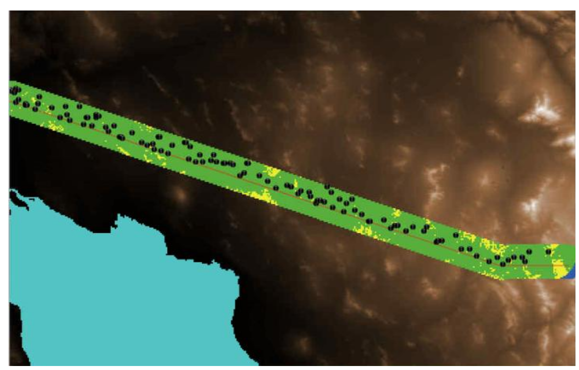
Figure 66: Summary of the Coverage as a Function of the Cost of the Single Type Systems Detection Architectures

In Figure 63, Table 20, Figure 64, and Figure 65, one can notice that the detection architectures composed of the systems with the largest detection ranges reach, and even surpass, the minimum coverage threshold of 90% with a final number of systems less than the maximum allowed value of 200 (*S_Max*). This is the case for the high cost radar, the medium cost radar, and the high cost camera detection architectures. These are also the detection architectures which cost the most given that they are composed of high performance sensor systems. On the contrary, the detection architectures featuring the low cost radar, the medium cost camera, and the low cost camera fail to satisfy the minimum coverage constraint of 90% although they are composed of a final number of systems equal to the maximum allowed value of 200 (*S_Max*). These detection architectures also correspond to lower resulting costs.

The observations above are rather straightforward when considering the dimensions of the problem and can be easily visualized in the NetLogo environment. Figure 67 depicts the actual detection coverage of the detection architectures resulting from the single type systems recursive optimization for the six types of detection systems considered (HCR, MCR, LCR, HCC, MCC, and LCC).

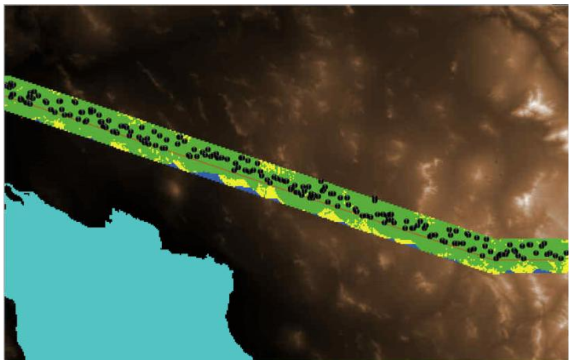


High Cost Radar Detection Architecture: 109 systems

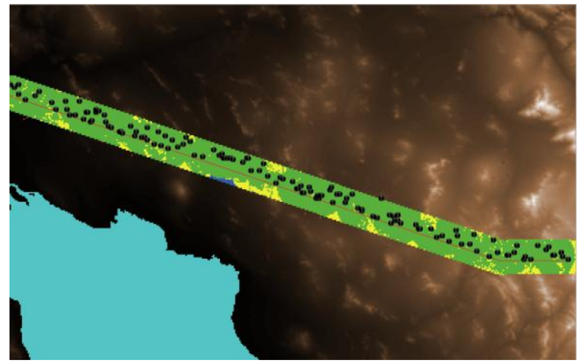


Medium Cost Radar Detection Architecture: 109 systems

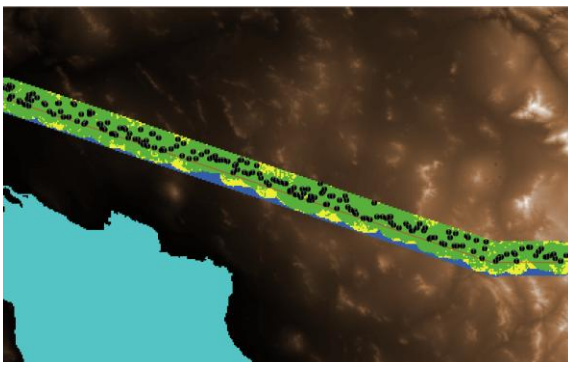
Figure 67: Actual Detection Coverage of the Optimized Detection Architectures Provided by the Single Type Systems Recursive Optimization Approach for Each Type of Detection System Visualized in the NetLogo Environment



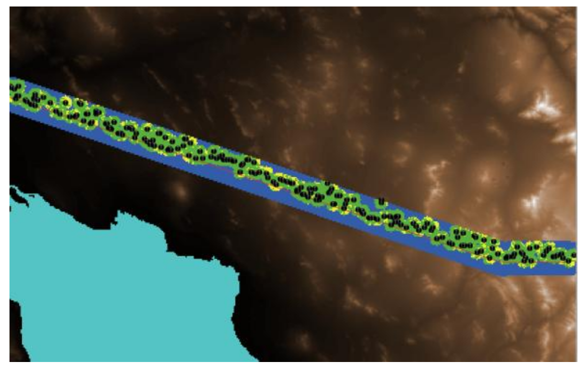
Low Cost Radar Detection Architecture:
200 systems



High Cost Camera Detection Architecture:
133 systems



Medium Cost Camera Detection Architecture:
200 systems



Low Cost Camera Detection Architecture:
200 systems

Figure 67: Actual Detection Coverage of the Optimized Detection Architectures Provided by the Single Type Systems Recursive Optimization Approach for Each Type of Detection System Visualized in the NetLogo Environment (Continued)

In Figure 67, the black dots are detection systems, the blue patches are terrain grids inside the detection band that are not covered by the sensor systems, the green patches are terrain grids inside the detection band that are within the range and in the line-of-sight of the sensor systems, and the yellow patches are terrain grids inside the detection band that are within the range of the sensor systems but that are out-of-sight. In this context, the detection coverage of the single type systems detection architectures results from the ratio of the green patches to the total number of patches inside the band of detection (blue patches + yellow patches + green patches). Consequently, the patches that cannot be seen by the detection architectures are the yellow patches and the remaining blue patches.

In the single type systems recursive optimization approach, the goal is to obtain detection architectures with the highest terrain coverage for the minimum cost. In Figure 66, this condition is obtained for the solution located at the upper left corner of the graph. This indeed corresponds to the detection architecture providing the maximum coverage at the minimum cost. In the case of the single type systems recursive optimization, this condition is obtained for a detection architecture composed entirely of high cost cameras. This was somewhat predictable due to their relatively low cost compared to all three radar types, and their relatively large range of detection compared to the other types of cameras. Nevertheless, a detection architecture composed of only high cost cameras is neither realistic nor completely appropriate for the original CBP mission. Therefore, it is necessary to mix and match the different types of detection systems and determine an architecture of distributed systems of various types that would provide the maximum coverage at the minimum cost.

Finally, from Figure 63 and Figure 67, it is possible to identify 29 common promising locations at which all six types of detection systems tend to be positioned in the corresponding optimized detection architectures provided by the single type systems recursive optimization. This set of 29 common positions is depicted in Figure 68 and the corresponding latitudes, longitudes, and position indices are summarized in Appendix M.

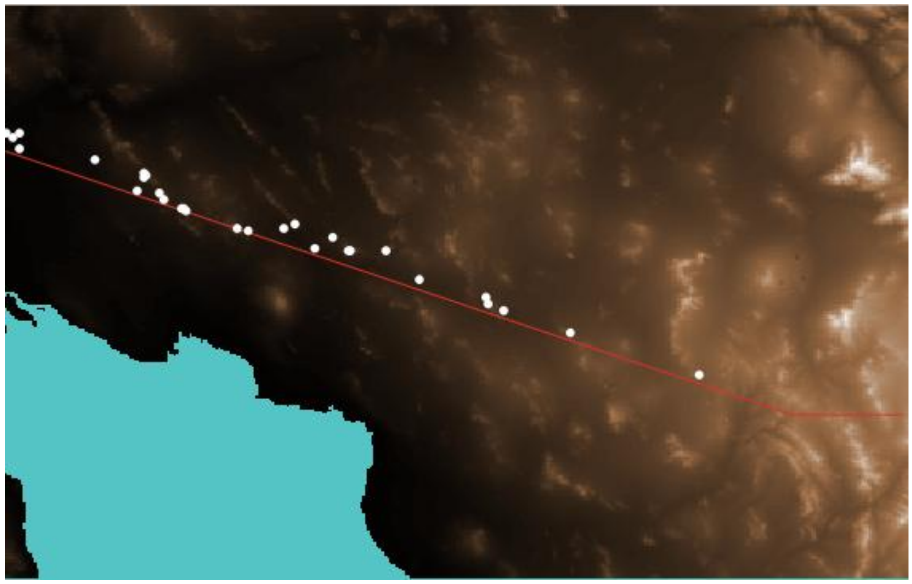


Figure 68: Common set of Promising Locations at Which Detection Systems are Preferentially Located in the Single Type Systems Recursive Optimization Approach Across all System Types

6.3.2. Multiple Type Systems Recursive Optimization

The multiple type detection systems recursive optimization approach is composed of the following steps:

1. The *Available_pos* variable is initialized to the complete list of most promising positions provided in Appendix I.
2. Starting from each of the 256 promising positions in turn, the following steps are performed:
    a. For the promising position considered in step 2, the type of the detection system associated with the highest coverage effectiveness value is determined from Appendix L. This step is notionally depicted in Figure 69

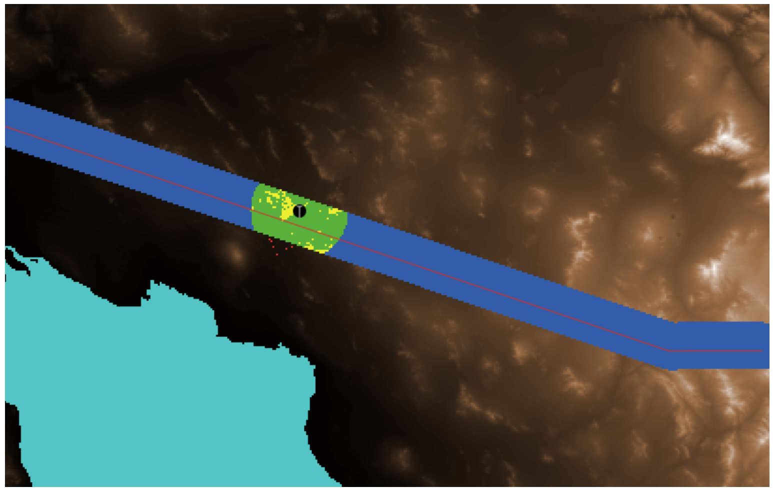


Figure 69: Recursive Optimization Approach With Multiple Types of Detection Systems – First Step

b. The *Sensor_list* variable is initialized to the index corresponding to the type of detection system obtained in step *a*.

c. The cost of the current detection architecture *Current_Architecture_Cost* is initialized to the cost of the detection system determined in step *a*.

d. The *Occupied_pos_list* variable is initialized to the index of the promising position considered in step 2.

e. The coverage of the current detection architecture *Current_Architecture_Coverage* is initialized to the coverage effectiveness value associated with the detection system determined in step *a*., at the promising position considered in step 2.

f. The *Occupied_pos* variable is initialized to the latitude, longitude, and index of the promising position considered in step 2.

g. The *Available_pos* variable is updated based on the *Occupied_pos* variable obtained in step *f*.

h. While the number of sensor systems in the current detection architecture is less than *S_Max* and/or the coverage of the current detection architecture is less than *Min_Cov* and/or the cost of the current detection architecture is not more than *Max_Cost*, a new detection system is added to the current detection architecture based on the following operations:

ii. For each type of detection system, located at each available promising position stored in the *Available_pos* variable, the coverage of the resulting detection architecture is calculated using the NetLogo environment. This enables taking into account the potential overlap in coverage of detection systems in the architecture.

iii. A set *S_candidate* of *nb_candidate* resulting detection architectures presenting the highest coverage values amongst all the candidate detection architectures obtained in step *ii* is selected to continue the optimization. This is notionally depicted in Figure 70. While the number of detection systems in the candidate detection architectures is less than *S_Max* and/or the coverage of the current detection architecture is less than *Min_Cov* and/or the cost of the current detection architecture is not more than *Max_Cost*, a nested recursive optimization, identical to the one described in this section, is performed on all the candidate architectures in the set *S_candidate* so as to recursively determine detection architectures presenting the highest coverage.

- For each candidate detection architecture in the set *S_candidate*, the index corresponding to the type of the newly added detection system in the candidate detection architecture and the promising position at which the new detection system will be located immediately result.

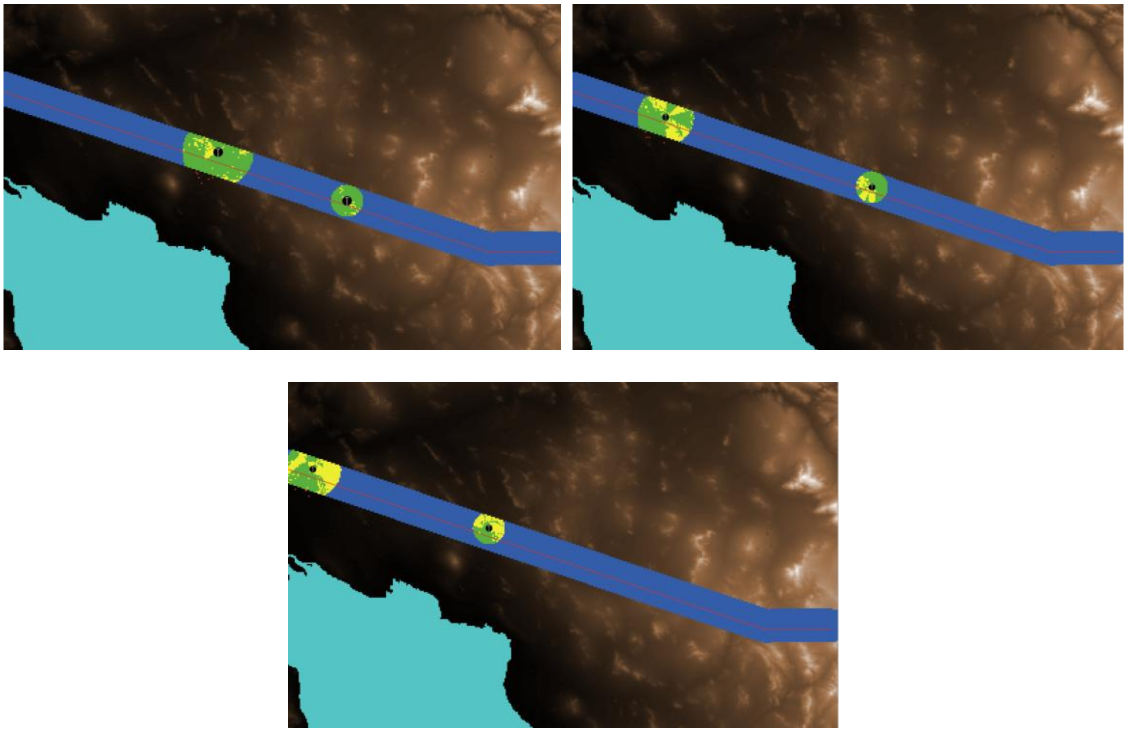


Figure 70: Recursive Optimization Approach With Multiple Types of Detection Systems – Pool of Candidate Detection Architectures (for *nb_candidate* = 3)

- For each candidate detection architecture in the set *S_candidate*, the *Sensor_list* variable is updated with the type of the detection system added to the corresponding detection architecture, obtained from step *iii*.
- For each candidate detection architecture in the set *S_candidate*, the cost *Current_Architecture_Cost* is updated with the cost of the detection system added to the corresponding detection architecture, obtained in step *iii*.
- For each candidate detection architecture in the set *S_candidate*, the *Occupied_pos_list* variable is updated with the index of the promising position obtained in step *iii*, at which the new detection system is located.
- For each candidate detection architecture in the set *S_candidate*, the coverage *Current_Architecture_Coverage* is updated to the coverage of the corresponding newly

generated detection architecture determined from steps *ii* and *iii*.

- For each candidate detection architecture in the set *S_candidate*, the *Occupied_pos* variable is updated with the latitude, longitude, and index of the promising position obtained in step *iii*, at which the new detection system is located.
- For each candidate detection architecture in the set *S_candidate*, the *Available_pos* variable is updated based on the *Occupied_pos* variable obtained in the previous step.
- Steps *ii* onward are then applied to each candidate detection architecture in the set *S_candidate*.

The aforementioned algorithm is run at each promising position identified in Figure 41 and Appendix I to generate a meaningful number of detection architectures to study. In this work, a number *nb_candidate* of 10 candidate detection architectures is carried over at each step of the nested optimizations. A maximum number of systems allowed in the architecture *S_Max* of 200 and a minimum coverage value *Min_Cov* of 70% were also specified. In this case, the constraint on the minimum coverage required for the total architecture has been relaxed compared to the previous study in order to consider the fact that multiple systems will be mixed together and positioned at locations chosen among a pre-defined set of promising positions. In this context, it is highly probable that the resulting architectures will not meet the 90% minimum coverage constraint contrary to single type systems detection architectures obtained in the previous case and composed of sensors with the largest detection ranges. Finally, the constraint on the maximum cost allowed *Max_Cost* was not specified. That is why, the total number of systems in the resulting detection architectures may be less than *S_Max* provided that the coverage of the said architectures is more than *Min_Cov*.

Figure 71 and Figure 72 show the graphs of the coverage as a function of the cost, and of the number of cameras as a function of the number of radars in the optimized detection architectures provided by the multiple type systems recursive optimization approach applied to the CBP optimization problem. In Figure 71, the black line represents the Pareto optimal detection architectures which will serve as a benchmark for the detection architectures provided by the evolutionary optimization of the original CBP optimization problem. They will be investigated further in a subsequent section.

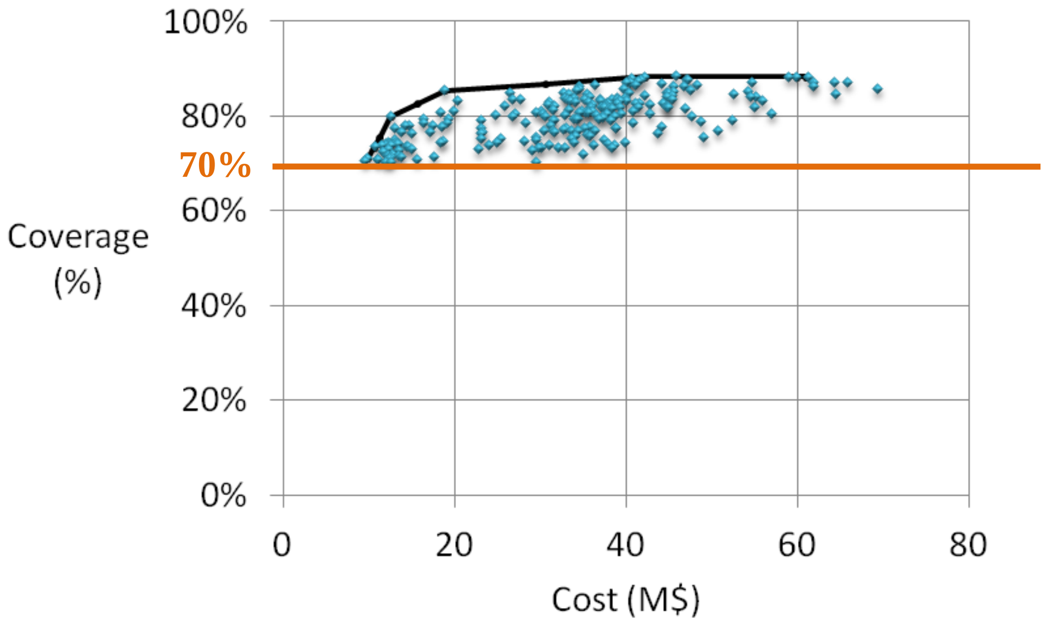


Figure 71: Graph of the Coverage as a Function of the Cost of the Optimized Detection Architectures Obtained From the Multiple Type Systems Recursive Optimization Approach

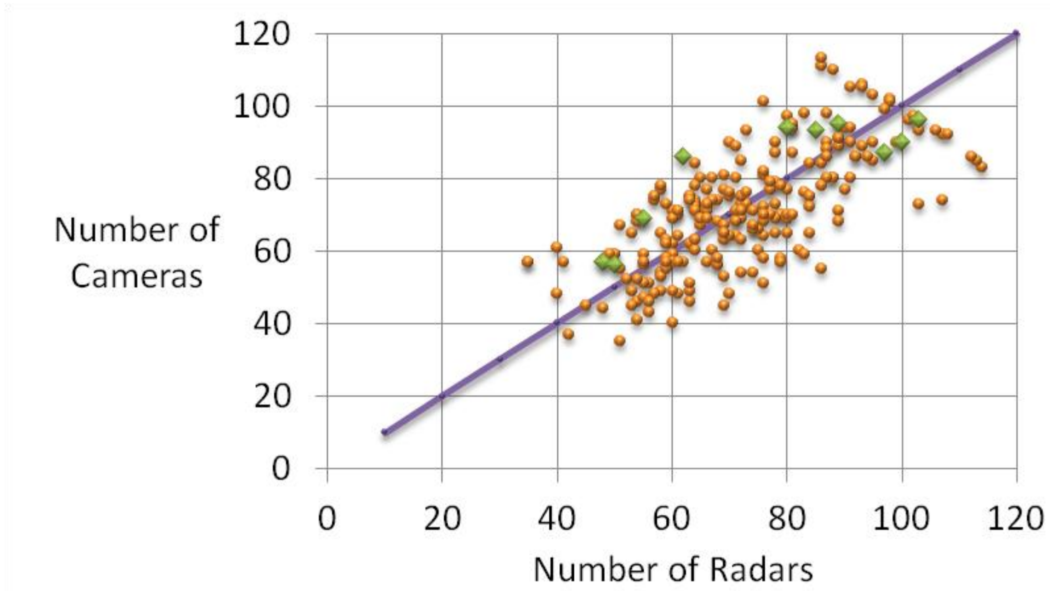
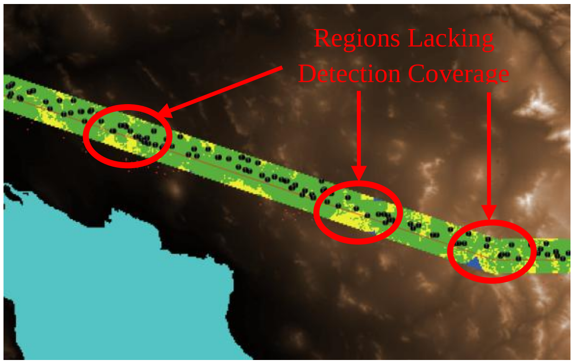


Figure 72: Graph of the Number of Cameras as a Function of the Number of Radars in the Optimized Detection Architectures Obtained From the Multiple Type Systems Recursive Optimization Approach

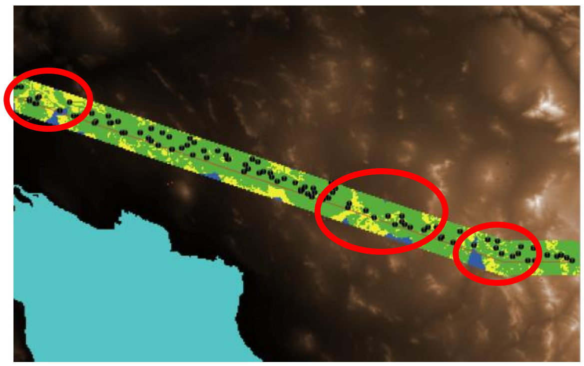
From Figure 72, one can notice that the optimized detection architectures do not tend to favor one type of detection system over the other. Indeed, the types of detection systems in the resulting set of detection architectures seem to be rather well distributed amongst radars and cameras.

In Figure 72, the green diamonds represent ten of the Pareto optimal detection architectures identified in Figure 71. The surface coverage of each of these ten Pareto detection architectures is depicted in the NetLogo environment in Figure 73.

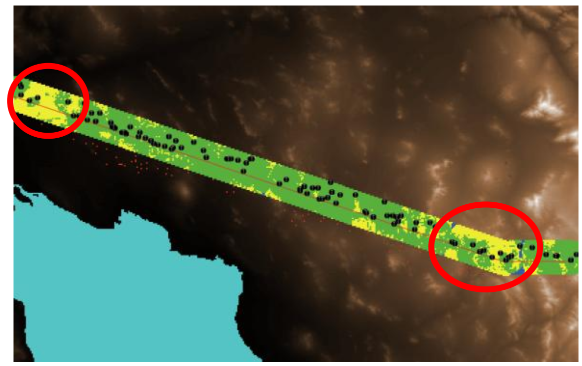


Architecture 1

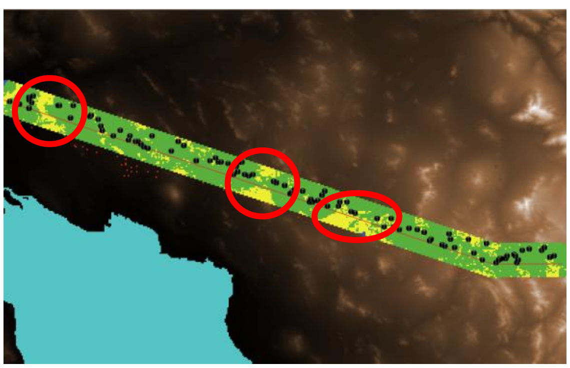
Architecture 2



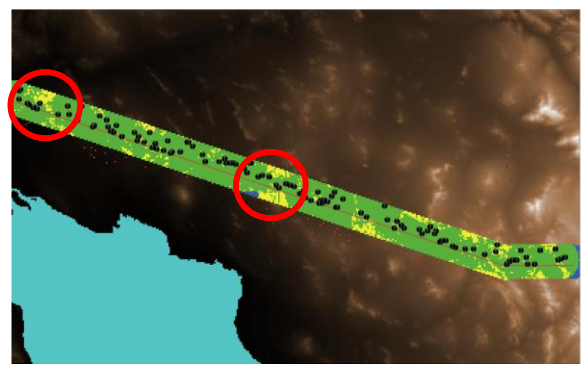
Architecture 3



Architecture 4

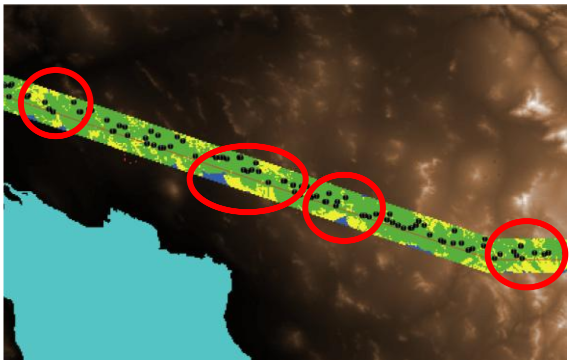


Architecture 5



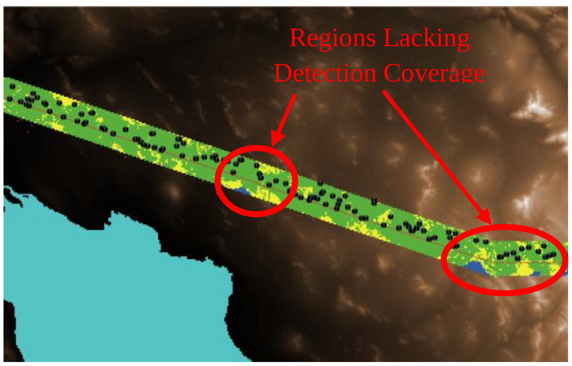
Architecture 6

Figure 73: Actual Detection Coverage of Ten of the Pareto Optimal Detection Architectures Provided by the Multiple Type Systems Recursive Optimization Approach Visualized in the NetLogo Environment

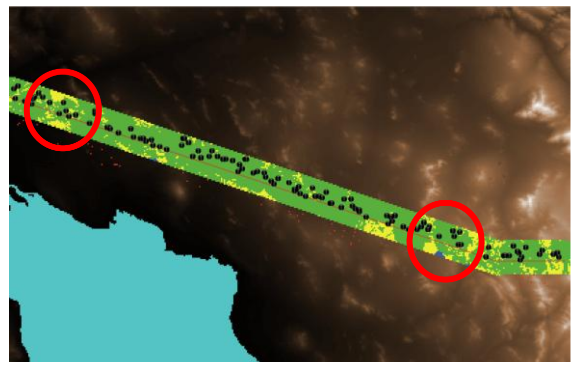


Architecture 7

Architecture 8



Architecture 9



Architecture 10

Figure 73: Actual Detection Coverage of Ten of the Pareto Optimal Detection Architectures Provided by the Multiple Type Systems Recursive Optimization Approach Visualized in the NetLogo Environment (Continued)

In Figure 73, the black dots are detection systems while the blue patches are terrain grids inside the detection band that are not covered by the sensor systems. The green patches are terrain grids inside the detection band that are within the range and in the line-of-sight of the sensor systems, while the yellow patches are terrain grids inside the detection band that are within the range of the sensor systems but that are not visible due to obstruction of their fields-of-view by terrain obstacles. In this context, the detection coverage of the multiple type systems detection architectures results from the ratio of the green area to the total area within the band of detection (blue + yellow + green patches). Consequently, the region not seen by any system in the detection architectures is composed of the yellow and blue areas. In Figure 73, one may notice that the detection coverage of the depicted architectures is rather well spread across the band of detection along the border. Nevertheless, some yellow and some blue areas where the detection coverage is limited may be identified. These are regions where detection systems transported by patrolling CBP agents may potentially be deployed and operated directly on mobile platforms to complete and enhance the detection capabilities of the fixed detection architectures obtained by the recursive approach. These regions are identified by red circles in Figure 73. Additionally, Figure 73 shows that the set of Pareto optimal detection architectures considered have very similar structures on the terrain. The detection systems tend to be preferentially located at the same best positions, where they have the highest coverage efficiency.

Table 21 summarizes the coverage and the cost corresponding to the ten Pareto optimal detection architectures considered previously, along with their total number of systems, their number of radars and their number of cameras.

Table 21: Properties of Ten of the Pareto Optimal Detection Architectures Provided by the Multiple Type Systems Recursive Optimization Approach When Applied to the CBP Optimization Problem

	Coverage (%)	Cost (M$)	Number of Systems	Number of Radars	Number of Cameras
Architecture 1	82.6	44.7	157	79	78
Architecture 2	80.8	32.7	164	86	78
Architecture 3	77.9	22.8	131	57	74
Architecture 4	76.8	44	119	60	59
Architecture 5	82.4	42.6	149	84	65
Architecture 6	87	39.9	157	78	79
Architecture 7	77.8	20.8	139	66	73
Architecture 8	82.2	32.2	178	89	89
Architecture 9	83.2	28	184	90	94
Architecture 10	84.9	41.4	170	86	84

Finally, the single type systems recursive optimization approach enabled identify a set of twenty nine common promising locations at which the six types of detection systems were preferentially located in the resulting detection architectures. In the multiple type systems recursive optimization approach however, all the two hundred and fifty six promising positions have been utilized by the detection systems in one architecture or the other. Therefore, it makes no real sense to try to determine a set of mostly used promising positions in this case. What is more relevant is to be able to compare the properties of the Pareto optimal solutions obtained with this recursive approach with those of the Pareto optimal solutions provided by the evolutionary optimization approach. This will be described subsequently.

6.3.3. Concluding Remarks on Benchmarking Detection Architectures Using Heuristic Recursive Optimization

The previous sections have detailed the development and the structured analysis of a heuristic recursive optimization scheme based on simple performance, cost and geometrical positioning rules. The goal of this study was to provide benchmark solutions for checking the accuracy of the Pareto optimal detection architectures provided by the evolutionary optimization of the original homeland security problem.

6.4. Evolutionary Optimization of the Customs and Border Protection Mission Scenario

The previous sections have enabled:

- The definition of a benchmark for the solution to the CBP optimization problem
- The investigation of appropriate sets of optimization parameters for both the modified GA and the modified PSO algorithm
- The comparison of the efficiency of the modified GA and of the modified PSO algorithm in solving test problems presenting similar properties as the original CBP optimization problem
- The selection of the modified PSO algorithm for solving the CBP mission scenario

With all the above in mind, the modified PSO algorithm can now be applied to the original CBP optimization problem in order to determine a portfolio of distributed detection system architectures composed of fixed sensors located at given positions on the terrain and providing the maximum coverage of the theater of operations at the minimum cost.

6.4.1. Proposed Optimization Approaches

In order to gauge the ability of the modified PSO algorithm to provide an accurate solution and to test its convergence efficiency for the CBP optimization problem, four different particle swarm optimization approaches are considered.

Full and global approach: in this first case, the detection systems may be placed at any positions in the band of detection. This corresponds to the full and global optimization approach in which the most promising locations are not implemented. This is depicted in Figure 74.

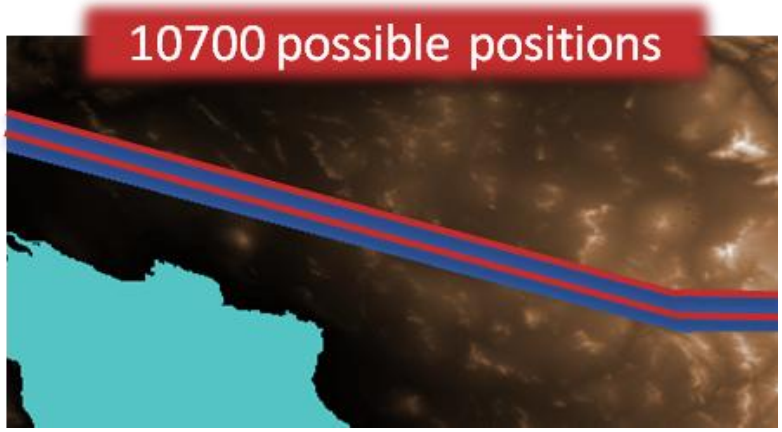


Figure 74: Full and Global Optimization Approach

Reduced and global approach: in this second case, the detection systems must be located at the pre-defined most promising positions in the band of detection. This case will be called the reduced and global optimization. In this approach, the resulting detection architectures presenting the best performances at the most affordable costs will be benchmarked against the solutions provided by the multiple type systems heuristic recursive approach. This is depicted in Figure 75.

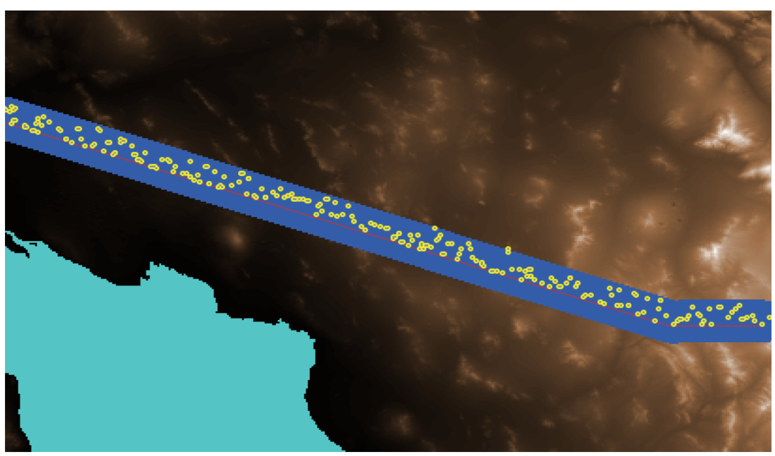


Figure 75: Reduced and Global Optimization Approach

Nested, full, and local approach: in this third case, the band of detection is divided into a number *Nb_Boxes* of equal sized boxes. The PSO algorithm is then used to optimize reduced detection architectures in each box. In this implementation, a number *S_Max* / *Nb_Boxes* of detection systems may be placed at any positions in the box considered and in the band of detection to generate reduced detection architectures providing the maximum coverage at the minimum cost for that box. The optimization algorithm is therefore successively applied locally to each box along the band of detection to determine locally optimized detection architectures. Then, it is assumed that the globally optimized detection architectures may be obtained by combining the first few "best" locally optimized detection architectures. This nevertheless requires some corrections in the resulting coverage due to overlap between boundary detection systems (located at the boundaries of the boxes). This approach corresponds to a nested optimization and will be referred to as the full and local optimization approach in which the most promising locations are not implemented. This is depicted in Figure 76.

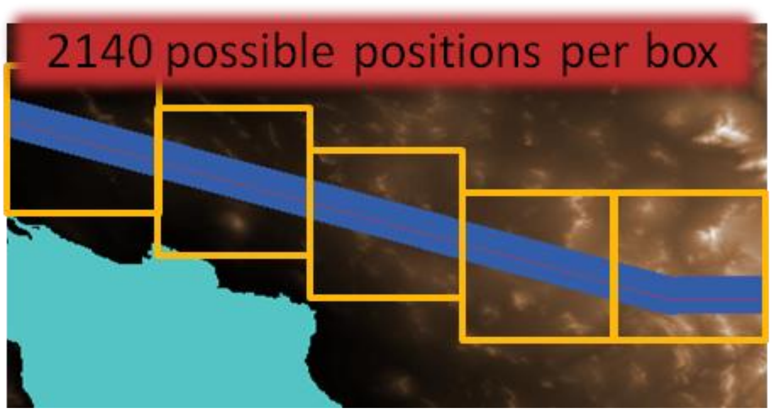


Figure 76: Nested, Full, and Local Optimization Approach

Nested, reduced and local approach: this fourth case is similar to the third case except that the detection systems in each box have to be located at the pre-defined most promising positions in the band of detection encompassed in the box of interest. This case is also a nested optimization and will be referred to as the reduced and local optimization. In this approach, the resulting combined detection architectures will be benchmarked against the solutions provided by the multiple type systems heuristic recursive optimization approach.

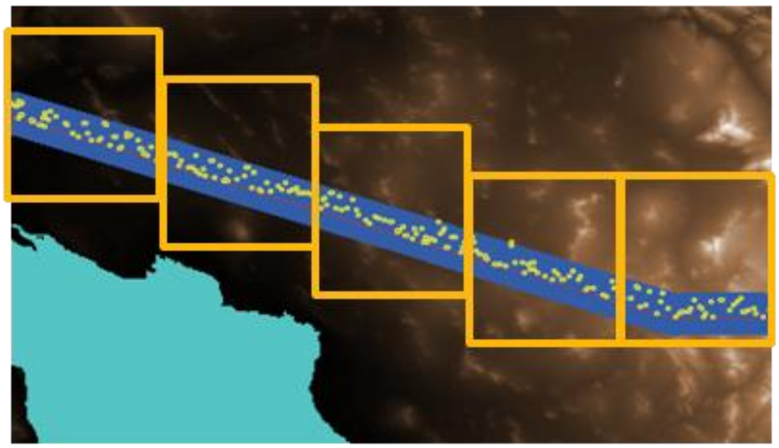


Figure 77: Nested, Reduced, and Local Optimization Approach

The aforementioned four variations of the particle swarm optimization algorithm have been run multiple times on the CBP optimization problem to compare their computational efficiency and their ability to provide reliable solutions. In all cases, the objective function is that constructed in Equation 5 with a weight factor $\alpha = 1$ unless otherwise specified. The results of the comparison analysis are provided in the next sections.

In this work, the four optimization approaches described above have been developed to analyze the ability of the proposed PSO algorithm to provide consistent solutions across multiple variants and to determine the most computationally efficient approach for the CBP optimization problem. However, this is not a limitation. Indeed, the four evolutionary optimization methods developed as part of this research may be used to perform various types of analyses. For instance, they may be employed to devise a detection architecture solution for a specific problem when no detection architecture presently exists. Starting from an existing detection architecture, the reduced and global optimization approach may enable to further enhance the detection capability of the detection architecture by appropriately modifying its structure. This may involve adding carefully selected detection systems at appropriate available promising positions, removing detection systems that do not provide sufficient coverage, or a combination of these two modifications. In the same context, the nested and local optimization methods may further optimize locally the various parts of an existing detection architecture to enhance its global operational effectiveness. Finally, all the aforementioned optimization approaches may be used as a means to design new detection technologies adapted to a specific need, that would provide additional coverage when and where required, whether it be for an existing or a newly designed detection architecture.

6.4.2. Convergence Analysis of the Proposed Optimization Approaches

The distribution function of the convergence times for the four evolutionary optimization approaches was investigated and did not exhibit any specific shape, which was expected. In order to draw more rigorous conclusions about which algorithm converges faster to a solution for the CBP optimization problem, and to quantify the spread in the convergence times, the means and the standard deviations statistics may be used. Figure 78 depicts the means and standard deviations of the convergence times for the four evolutionary optimization approaches across multiple runs.

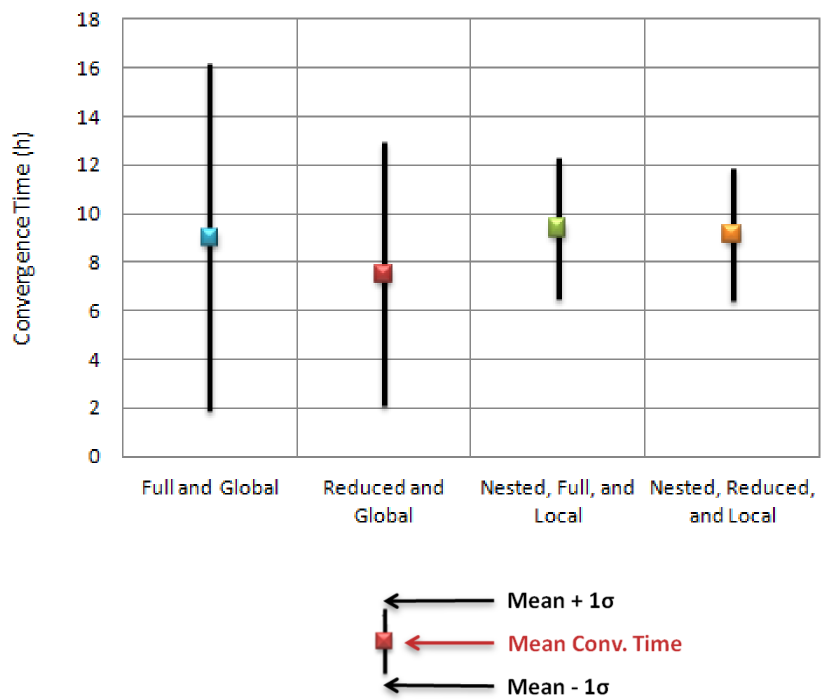


Figure 78: Plot of the Means and of the One-Sigma Boundaries of the Convergence Times for the Four Evolutionary Optimization Approaches Across Multiple Runs

Figure 78 shows that the reduced and global optimization approach, in which the set of promising positions inside the band of detection is pre-specified, converges the most rapidly on average. Nevertheless, the one-sigma standard deviation boundaries for this case are wider than those for the local optimization approaches, although they are smaller than those for the full and global optimization case. This means that, although the local optimization approaches tend to take more time to converge on average than the global optimization approaches, they are more consistent in terms of computational speed. This may be due to the fact that performing many local optimizations ends up being more computationally efficient than performing one global optimization.

When studying the convergence times for the four evolutionary algorithms considered in this research, it is assumed that a "good enough", yet global, solution to the original CBP optimization problem has been obtained. The solution of interest here is the one that covers the whole extent of the border, and not a series of local solutions that locally cover successive pieces of the border. In this context, it is obvious that the global optimization approaches provide global solutions. This is not the case for the local optimization approaches. Indeed, the solutions provided by the local optimization algorithms after convergence are locally optimized detection architectures, and not global detection architecture solutions to the original CBP optimization problem. Therefore, in those cases, it is necessary to account for the additional post-processing time required to recombine the locally optimized detection architectures into complete architectures, and to correct the resulting coverage for any overlap. This necessarily increases both the mean of the true convergence times and the one-sigma standard deviation boundaries of the local optimization cases.

Finally, it is worth noticing from Figure 78 that the reduced optimization approaches (global and local) are computationally more efficient at finding solutions to the corresponding optimization problem compared to their respective full approaches. This is because in the reduced optimization algorithms, the most promising positions at which detection systems must be located in the final detection architectures are already specified by the analyst, while in the full optimization approaches, detection systems may be located at any feasible positions inside the band of detection. This means that the full optimizations have more dimensionality, which translates into more positions to try, and thus more time to converge.

In this context, it is worth comparing more closely the reduced optimization approaches. Figure 79 shows the means and standard deviations of the convergence times for the reduced evolutionary optimization approaches across multiple runs.

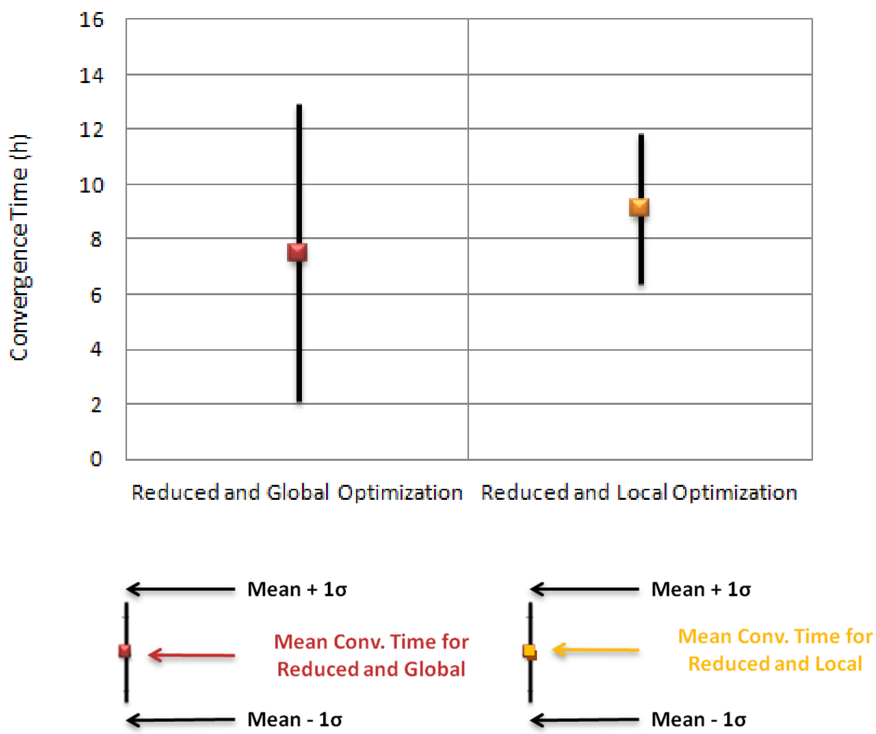


Figure 79: Plot of the Means and of the One-Sigma Boundaries of the Convergence Times for the Reduced Evolutionary Optimization Approaches Across Multiple Runs

From Figure 79, one may notice that although the reduced and global optimization seems to be converging faster on average than the reduced and local optimization, the latter seems to converge more consistently to a solution. Indeed, the spread in the convergence time is smaller for the reduced and local optimization approach. Nevertheless, as mentioned previously, the reduced and global optimization approach provides a globally optimized detection architecture. On the contrary, the reduced and local optimization approach provides locally optimized detection architectures that need to be recombined into a global detection architecture to obtain a solution structurally similar to the ones provided by the corresponding global approach. This adds post-processing time to the actual "convergence" time of the reduced and local optimization. Ultimately, when the additional post-processing time required to obtain global solutions is taken into account, the reduced and local optimization approach converges much slower on average than its global counterpart. This is notionally depicted in Figure 80.

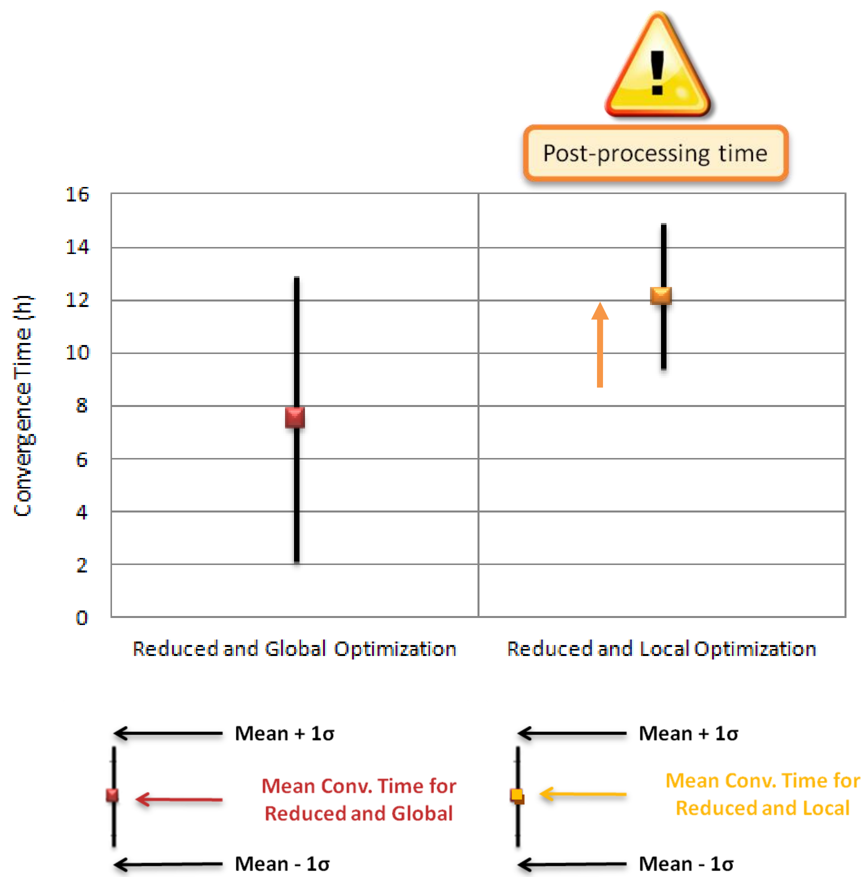


Figure 80: Plot of the Means and of the One-Sigma Boundaries of the Convergence Times for the Reduced Evolutionary Optimization Approaches Across Multiple Runs, With Added Post-Processing Time for the Local Approach

Eventually, the reduced and global optimization approach converges more efficiently than the reduced and local optimization approach when considering post-processing time, and is thus the most computationally adapted to the CBP problem. The solutions provided in this case are analyzed in subsequent sections.

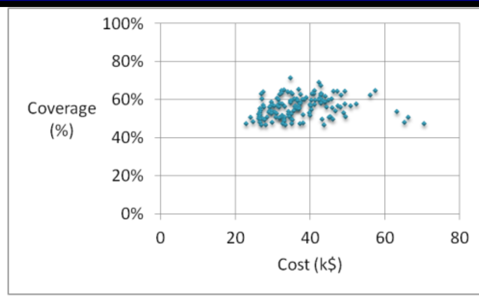
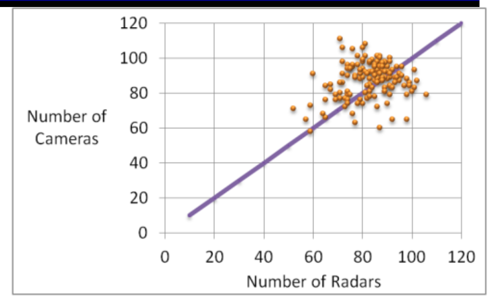
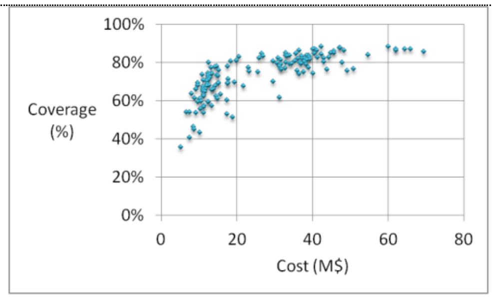
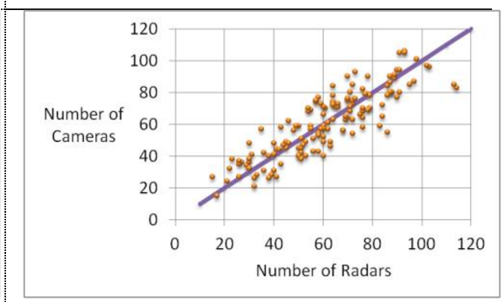
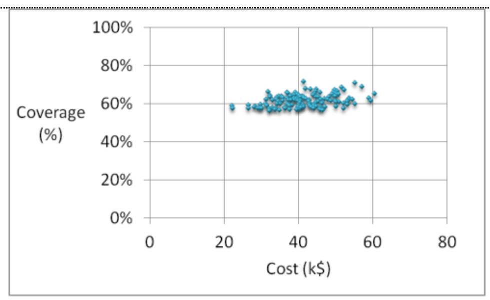
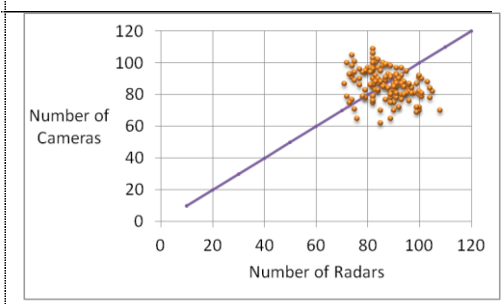
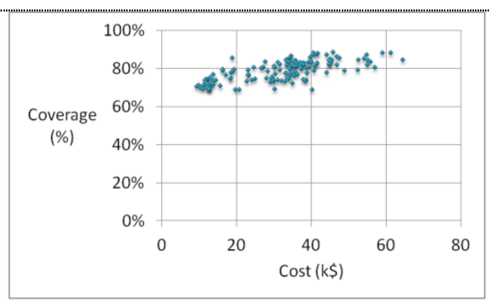
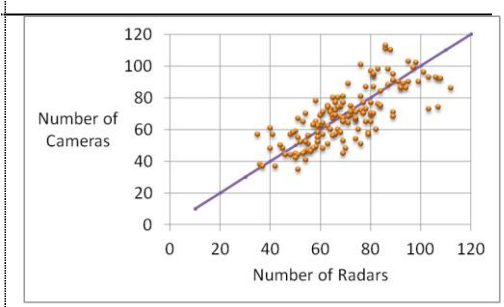
6.4.3. Properties of the Solutions Provided by the Proposed Optimization Approaches

In this section, the goal is to study the performance of the proposed four particle swarm optimization algorithms at providing solutions to the original CBP optimization problem. In order to do so, it is interesting to study the properties of a set of detection architectures that are obtained over the last few hundreds of generations before convergence of the algorithms, and that present the maximum coverage to cost ratios. These architectures will be called “best ratio” detection architectures. The results of the analysis may be summarized in two types of graphs:

- One depicting the coverage as a function of the cost of the set of detection architectures mentioned above
- One displaying the number of cameras in the final detection architectures as a function of the number of radars

Table 22 provides the resulting graphs for the four types of evolutionary optimization approaches described in the previous sections.

Table 22: Graphs of the Coverage as a Function of the Cost, and of the Number of Cameras as a Function of the Number of Radars in the set of "Best Ratio" Detection Architectures for the Four Evolutionary Optimization Approaches Considered

	Coverage as a Function of Cost for the set of "Best Ratio" Detection Architectures	Number of Cameras as a Function of Number of Radars in the set of "Best Ratio" Detection Architectures
Full and Global Optimization		
Reduced and Global Optimization		
Nested, Full, and Local Optimization		
Nested, Reduced, and Local Optimization		

From Table 22, one may notice that the global and the local reduced optimization approaches provide detection architectures with better coverage to cost ratios than the global and the local full optimization approaches. This is most certainly because the promising positions at which the detection systems have the highest coverage efficiency are specified. This undoubtedly orients the search for the most effective detection architectures in the reduced optimization approaches. In the full optimization approaches however, the search algorithms have no guidance about the performance of the detection systems at all the feasible positions inside the band of detection. Therefore, they must perform by trial and error. In this context, the search is not as efficient, and its ability to find effective detection architectures is diluted by the combinatorial nature of the problem. From Table 22, it is also evident that none of the reduced optimization approaches implemented in this study tends to favor one type of detection system over the other. Indeed, the types of detection systems in the resulting set of "best ratio" architectures seem to be rather well distributed amongst radars and cameras. One can notice though that the full evolutionary optimization algorithms (global and local) tend to diverge towards one or the other type of detection system as the total number of systems in the detection architectures increases. Again, this may come from the fact that the search for detection architecture solutions is more random in those cases than in the reduced optimization algorithms due to the lack of preliminary knowledge about the most promising positions.

Finally, the distribution function of the total number of systems in the set of "best ratio" detection architectures provided by the four evolutionary optimization approaches was investigated and did not exhibit any specific shape, which was expected. In order to compare the properties of the "best ratio" detection architectures, the means and the standard deviations statistics may be used. Figure 81 gives the means and one-sigma standard deviation boundaries for the total number of systems in the set of "best ratio" detection architectures provided by the four evolutionary optimization approaches considered in this work.

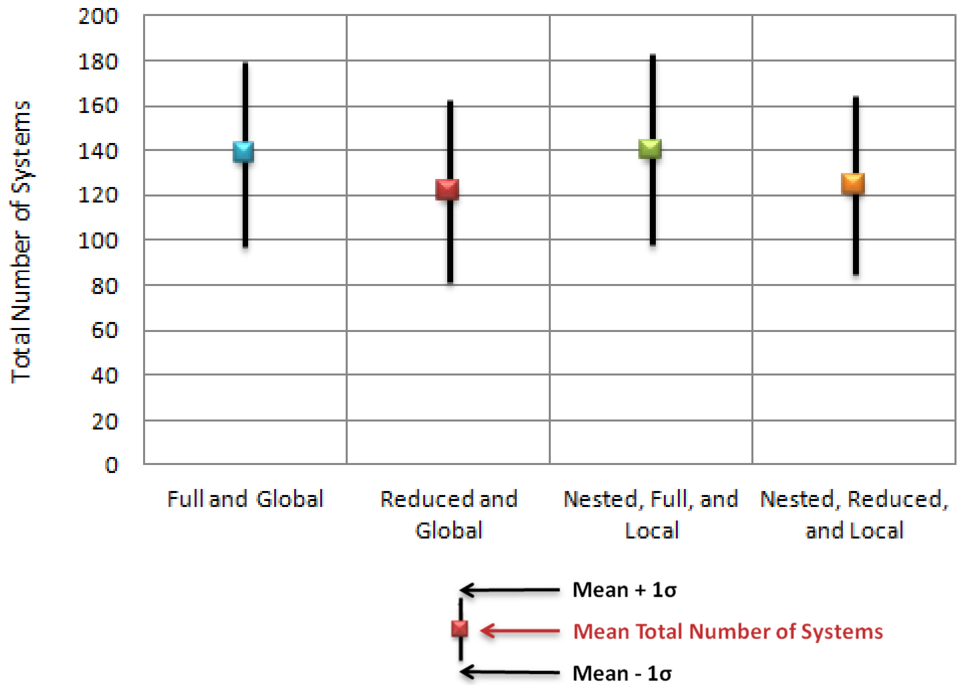


Figure 81: Plot of the Means and of the One-Sigma Boundaries of the Total Number of Systems in the set of "Best Ratio" Detection Architectures Provided by the Four Evolutionary Optimization Approaches Considered

Figure 81 shows that, on average, the global and the local reduced optimization approaches yield detection architectures with slightly fewer detection systems than the global and the local full optimization approaches. Once again, this feature may be explained by the fact that the most promising locations are specified in those cases, while the full optimization approaches have to explore all the feasible positions inside the band of detection as they search for a solution to the problem. This obviously decreases their performance. In this context, and using the information provided in Table 22, the reduced optimization algorithms are able to find detection architectures presenting better coverage and cost characteristics with slightly fewer systems than the full optimization algorithms. Furthermore, as demonstrated in the previous section, among the reduced optimization approaches, the global one is the most computationally adapted to the CBP problem. The solutions provided in this case are subject to further investigations in the next section.

6.4.4. Analysis of Pareto Efficient Solutions Provided by the Reduced and Global Optimization Approach for the Customs and Border Protection Optimization Problem

Consider Figure 82 depicting the set of "best ratio" detection architectures obtained from the reduced and global optimization approach applied to the CBP mission scenario.

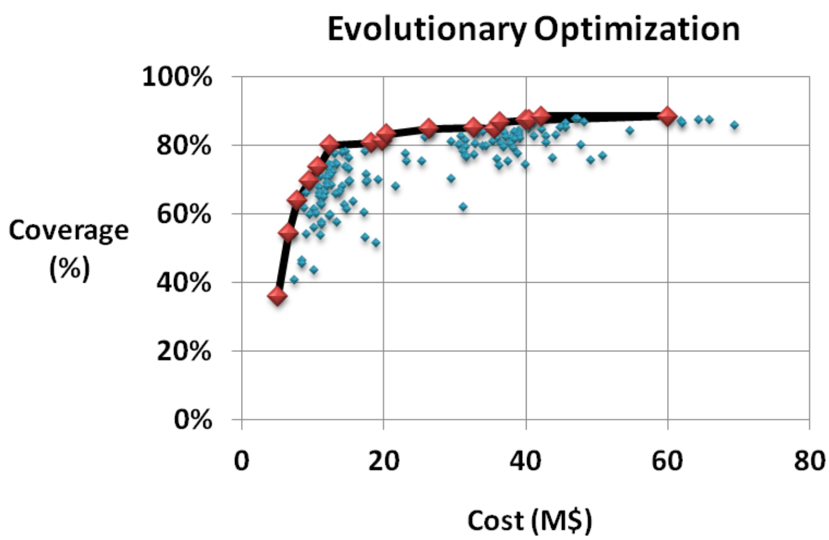
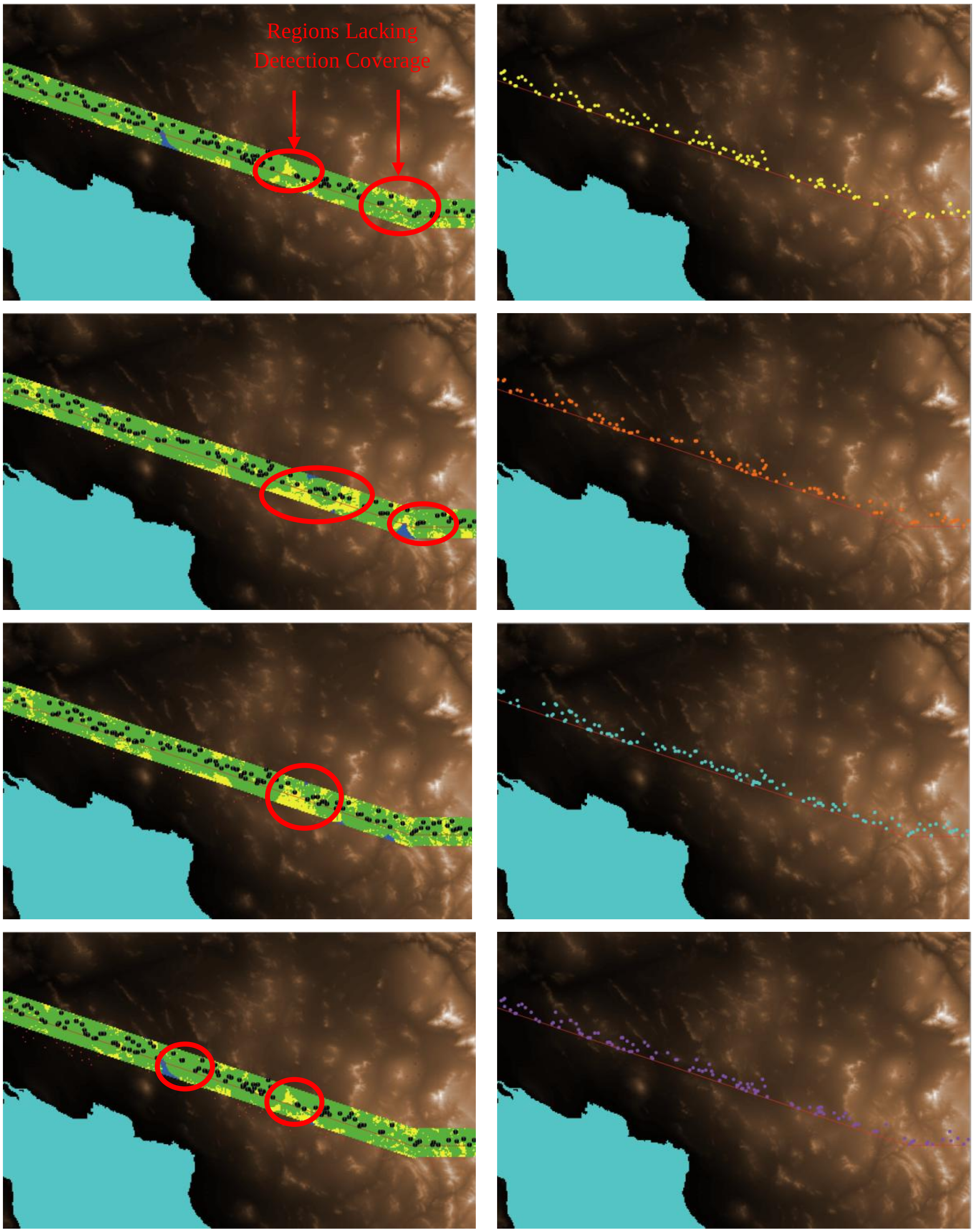


Figure 82: Plot of the Coverage to the Cost of the set of "Best Ratio" Detection Architectures for the Reduced and Global Optimization Algorithm (with Specification of the Most Promising Positions)

From Figure 82, a Pareto front of detection architectures providing the maximum coverage for different resulting costs may be identified, as illustrated by the black line. Some of these Pareto detection architectures, corresponding to different customer preferences are highlighted in red in Figure 82. This set of "coverage-to-cost Pareto efficient" detection architectures would correspond to increasing preference in detection coverage and thus decreasing interest in resulting cost. These Pareto detection architectures may be visualized in the NetLogo environment. This is depicted in Table 23.

Table 23: Illustration of the Coverage and of the Structure of the "Coverage-to-Cost Pareto Efficient" Detection Architectures Obtained From the Reduced and Global Evolutionary Optimization Approach Applied to the CBP Optimization Problem

Detection Coverage	*Structure of the Detection Architecture*



On the rightmost pictures in Table 23, one can notice that the “coverage-to-cost Pareto efficient” detection architectures considered have very similar structures on the terrain. The detection systems tend to be preferentially located at the same best positions, where they have the highest coverage efficiency.

On the leftmost figures in Table 23, the black dots are detection systems while the blue patches are terrain grids inside the detection band that are not covered by the sensor systems. The green patches are terrain grids inside the detection band that are within the range and in the line-of-sight of the sensor systems, while the yellow patches are terrain grids inside the detection band that are within the range of the sensor systems but that are not visible due to obstruction of their fields-of-view by terrain obstacles. In this context, the detection coverage of the detection architectures results from the ratio of the green area to the total area within the band of detection (blue + yellow + green patches). Consequently, the region not seen by any system in the detection architectures is composed of the yellow and blue areas. In Table 23, one can notice that the detection coverage of the depicted Pareto efficient architectures is rather well spread across the band of detection along the border. Nevertheless, some yellow and some blue areas where the detection coverage is limited may be identified. These are regions where detection systems transported by patrolling CBP agents may potentially be deployed and operated directly on mobile platforms to complete and enhance the detection capabilities of the fixed detection architectures depicted in Table 23. These regions are identified by red circles on the leftmost figures of Table 23.

Table 24 summarizes the coverage and the cost corresponding to the “coverage-to-cost Pareto efficient” detection architectures, along with their total number of systems, their number of radars and their number of cameras.

Table 24: Properties of the "Coverage-to-Cost Pareto Efficient" Detection Architectures Obtained From the Reduced and Global Evolutionary Optimization Approach Applied to the CBP Optimization Problem

	Coverage (%)	Cost (M$)	Number of Systems	Number of Radars	Number of Cameras
Architecture 1	80	12.5	124	55	69
Architecture 2	83.2	20.4	184	90	94
Architecture 3	86.4	36.4	177	87	90
Architecture 4	88.3	42.2	184	97	87

6.4.5. Checking the Accuracy of the Solutions Provided by the Reduced and Global Optimization Approach With the Benchmark Solutions Obtained from the Heuristic Recursive Approach

Consider the set of "best ratio" detection architectures provided by the evolutionary optimization and depicted in Table 22 and Figure 82 for the reduced and global optimization approach.

The distribution function of the total number of systems, the number of radars, the number of cameras, the coverage (in %) and the cost (in M$) of the set of "best ratio" detection architectures was investigated and did not exhibit any specific shape, which was expected. In order to compare the properties of these detection architectures, the means and the standard deviations statistics may be used. Table 25 gives the means and the one-sigma standard deviation boundaries for the total number of systems, the number of radars, the number of cameras, the coverage (in %) and the cost (in M$) for this set of "best ratio" detection architectures. This is illustrated in Figure 83, where the horizontal index represents the parameter considered (1 is the total number of systems in the architecture, 2 is the number of radars, 3 is the number of cameras, 4 is coverage, and 5 is cost), and the vertical axis is either a number (for the number of systems, radars, and cameras), or a percentage (for the coverage), or a currency (M$ for the cost).

Table 25: Means and Standard Deviations of the Properties of the set of "Best Ratio" Detection Architectures Provided by the Reduced and Global Evolutionary Optimization Algorithm

	Mean	Standard Deviation
Number of Systems	121.92	40.39
Number of radars	61.16	21.28
Number of Cameras	60.76	20.54
Coverage (%)	73.46	11.15
Cost (M$)	26.22	15.26

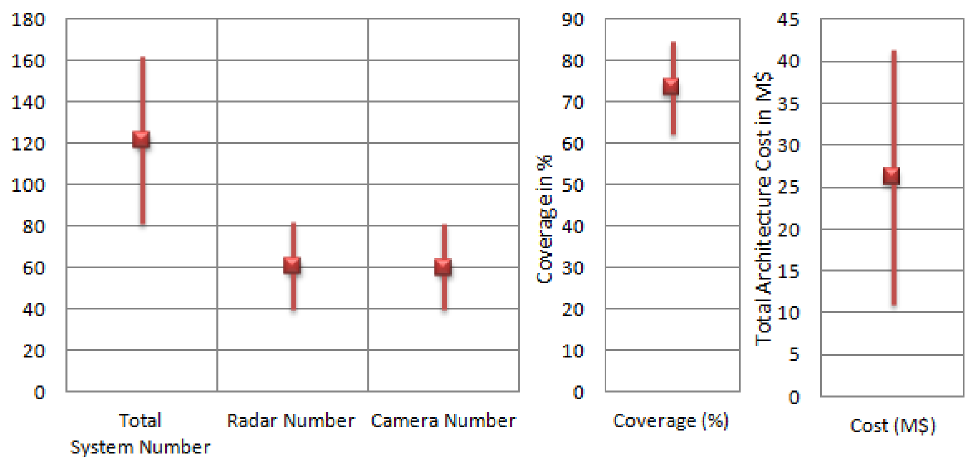


Figure 83: Plot of the Means and of the One-Sigma Boundaries of the Properties of the set of "Best Ratio" Detection Architectures Provided by the Reduced and Global Evolutionary Optimization Algorithm

Consider the set of benchmark detection architectures provided by the multiple type systems recursive optimization approach depicted in Figure 71.

The distribution function of the total number of systems, the number of radars, the number of cameras, the coverage (in %) and the cost (in M$) of the set of benchmark detection architectures was investigated and did not exhibit any specific shape, which was expected. In order to compare the properties of these detection architectures, the means and the standard deviations statistics may be used. Table 26 gives the means and the one-sigma standard deviation boundaries for the total number of systems, the number of radars, the number of cameras, the coverage (in %) and the cost (in M$) for the benchmark detection architectures provided by the recursive optimization approach. This is illustrated in Figure 84, where the horizontal index represents the parameter considered (1 is the total number of systems in the architecture, 2 is the number of radars, 3 is the number of cameras, 4 is coverage, and 5 is cost), and the vertical axis is either a number (for the number of systems, radars, and cameras), or a percentage (for the coverage), or a currency (M$ for the cost).

Table 26: Means and Standard Deviations of the Properties of the Benchmark Detection Architectures Provided by the Multiple Type Systems Recursive Optimization Algorithm

	Mean	Standard Deviation
Number of Systems	137.48	40.81
Number of Radars	69.17	21.58
Number of Cameras	68.31	20.98
Coverage (%)	79.61	4.66
Cost (M$)	34.12	12.1

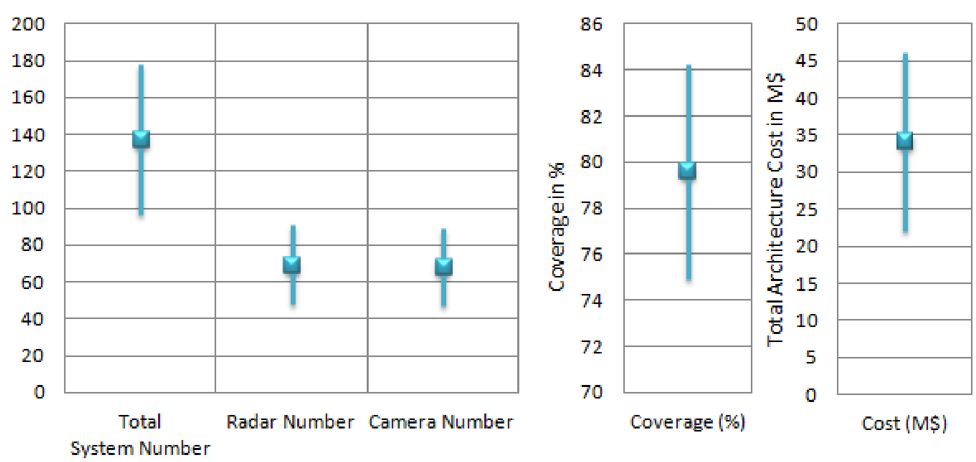


Figure 84: Plot of the Means and of the One-Sigma Boundaries of the Properties of the Benchmark Detection Architectures Provided by the Multiple Type Systems Recursive Optimization Algorithm

Figure 85 plots the means and the one-sigma standard deviation boundaries for the total number of systems, the number of radars, the number of cameras, the coverage (in %) and the cost (in M$) for both the set of "best ratio" detection architectures provided by the reduced and global evolutionary algorithm, and the benchmark detection architectures provided by the heuristic recursive optimization algorithm. In Figure 85, the horizontal index represents the parameter considered (1 is the total number of systems in the architecture, 2 is the number of radars, 3 is the number of cameras, 4 is coverage, and 5 is cost), and the vertical axis is either a number (for the number of systems, radars, and cameras), or a percentage (for the coverage), or a currency (M$ for the cost).

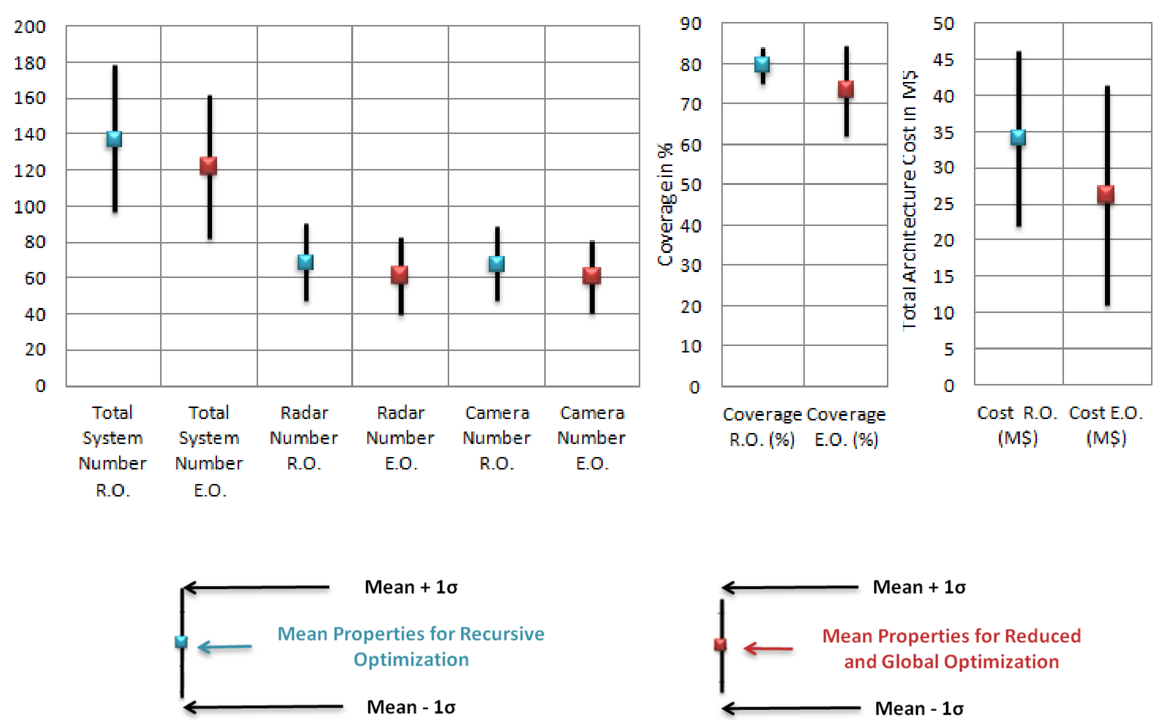


Figure 85: Comparison of the Means and of the One-Sigma Boundaries of the Properties of the Detection Architectures Provided by the Evolutionary Optimization (E.O.) and the Recursive Optimization (R.O.) Algorithms

Figure 85 shows that, on average, the "best ratio" detection architectures given by the evolutionary optimization approach are composed of a smaller number of systems compared to the detection architectures resulting from the recursive optimization approach. As a consequence, they also contain fewer radars and cameras comparatively. However, the standard deviation of the average total number of systems, number of radars, and number of cameras are almost identical in both optimization cases. This is rather encouraging and shows the consistency of the results between both types of optimization approaches. Additionally, given that the “best ratio” detection architectures involve fewer systems than the detection architectures provided by the recursive optimization, the cost of the said detection architectures is also lower. A similar causality effect between the total number of systems in the architecture and the resulting detection coverage is not evident however. Indeed, both the general structure of the detection architecture and the topography of the theater of operations come into play in this case. Nevertheless, everything else being equal, it may be expected that the detection coverage of the “best ratio” detection architectures obtained with the evolutionary optimization will also be smaller than that of the detection architectures resulting from the recursive optimization. This is confirmed in Figure 85. In addition, the standard deviation in the detection coverage of the architectures provided by the particle swarm optimization is more than twice that for the detection architectures obtained from the heuristic optimization. This comes from the evolutionary nature of the former algorithm which makes it less predictable than the entirely recursive heuristic approach. Finally, although the standard deviations in the architecture costs are rather similar for both approaches, it is relatively larger for the evolutionary optimization where it accounts for a little more than half of the corresponding mean value compared to about the third of the associated mean value for the recursive optimization.

To conclude, one may confidently say that the reduced and global particle swarm optimization algorithm does a good job at finding detection architectures that present similar characteristics as the benchmark solutions obtained with the recursive optimization algorithm. A detailed study of the Pareto fronts of optimal detection architectures provided by both optimization approaches shows that they have very similar shapes and locations. This is illustrated in Figure 86. The properties of the Pareto optimal detection architectures depicted in red in Figure 86 for both optimization cases are summarized in Table 27.

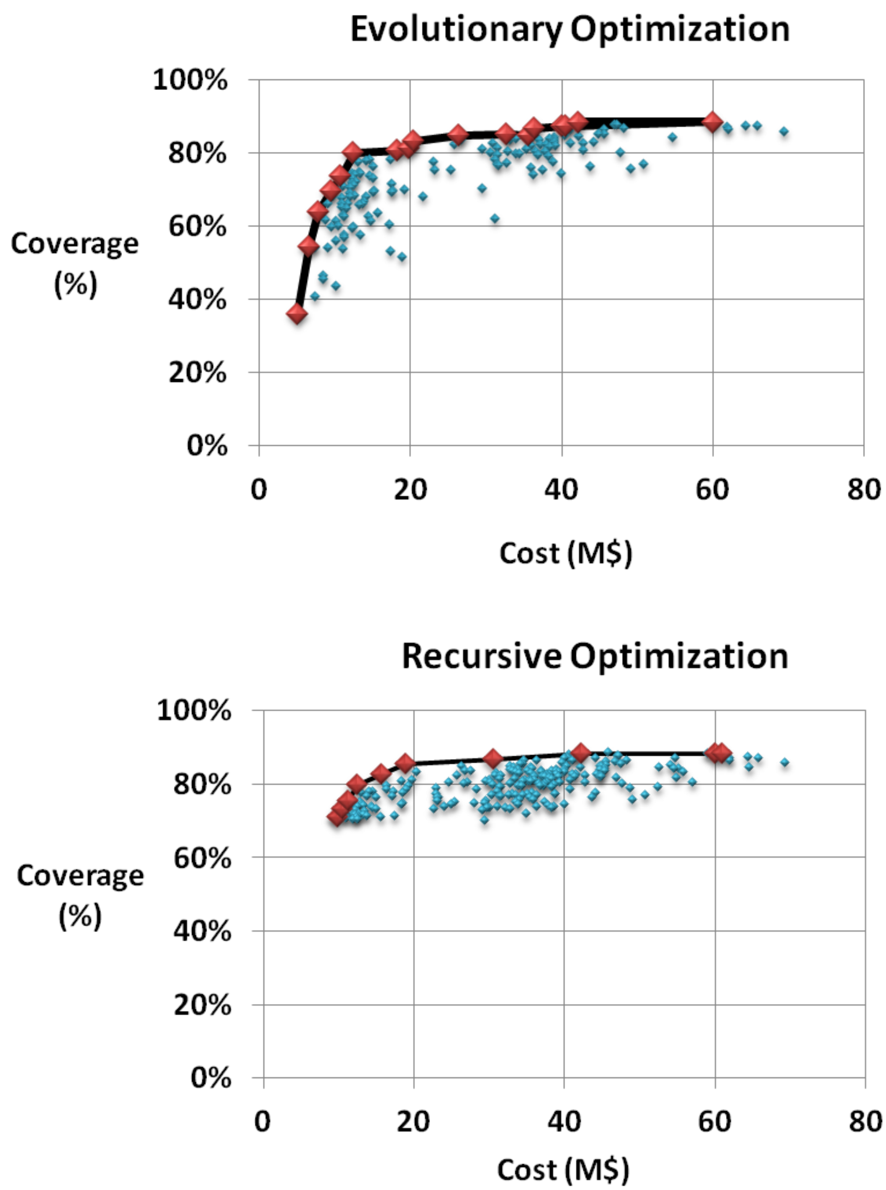


Figure 86: Graphs of the Coverage as a Function of the Cost, and of the Pareto Optimal Detection Architectures Obtained with the Evolutionary Approach (Top) and with the Recursive Approach (Bottom)

Table 27: Summary of the Properties of the Pareto Optimal Detection Architectures Obtained with the Evolutionary Approach (First) and with Recursive Approach (Next)

	Number of Systems	Number of Radars	Number of Cameras	Coverage (%)	Cost (M$)
PARETO OPTIMAL DETECTION ARCHITECTURES FOR THE EVOLUTIONARY OPTIMIZATION APPROACH	199	103	96	88.3	60
	168	88	80	87.6	47
	180	95	85	87.2	40.6
	157	78	79	87.1	40
	171	91	80	84.9	32.8
	131	61	70	84.7	26.4
	166	73	93	84.6	35.6
	158	76	82	81.1	19.9
	147	72	75	80.6	18.4
	124	55	69	79.9	12.5
	Number of Systems	Number of Radars	Number of Cameras	Coverage (%)	Cost (M$)
PARETO OPTIMAL DETECTION ARCHITECTURES FOR THE RECURSIVE OPTIMIZATION APPROACH	200	98	102	88.5	45.9
	197	101	96	88.3	59.1
	198	95	103	88.2	61.3
	185	95	90	88	40.7
	181	91	90	86.3	34.7
	184	106	78	85.4	17.6
	184	56	128	85.3	18.8
	145	74	71	79.4	16.4
	138	63	75	76.6	13.8
	122	59	63	74	12.4

Thus, it may be assumed that the results provided by the particle swarm optimization approach for the original CBP optimization problem are reliable.

6.4.6. Concluding Remarks on the Optimization of Detection Architectures for the Customs and Border Protection Mission Scenario

The previous sections have detailed the careful and transparent examination of the performance and the accuracy of the modified evolutionary optimization algorithm at determining reliable detection architectures for the homeland security scenario of interest. This step addressed the solutions benchmarking and accuracy checking research question, and validated the corresponding hypothesis. Finally, this last step further addressed the modeling, simulation, and optimization environment research question.

CHAPTER VII

IMPLEMENTATION – SIMULATION AND ANALYSIS OF A DETECTION ARCHITECTURE FOR A PROOF-OF-CONCEPT SCENARIO

The previous sections showed that the reduced and global optimization approach does a good job at finding accurate detection architecture solutions for the CBP mission scenario. With that in mind, one may now study the operational effectiveness of detection architectures in the modeling and simulation environment by simulating them under operational scenarios of interest. The following sections provide an example of study of the operational effectiveness of a "coverage-to-cost Pareto efficient" detection architecture provided by the modified particle swarm optimization algorithm.

7.1. Simulation of a Fixed Detection Architecture for the Customs and Border Protection Scenario

Consider the "coverage-to-cost Pareto efficient" detection architecture depicted as the red circle in Figure 87, and described in Table 28. This detection architecture represents a good compromise between coverage and cost according to the author, and might be the choice of a decision maker expecting bang for the buck. For now, this detection architecture is only composed of fixed detection systems positioned on the terrain at their optimized positions.

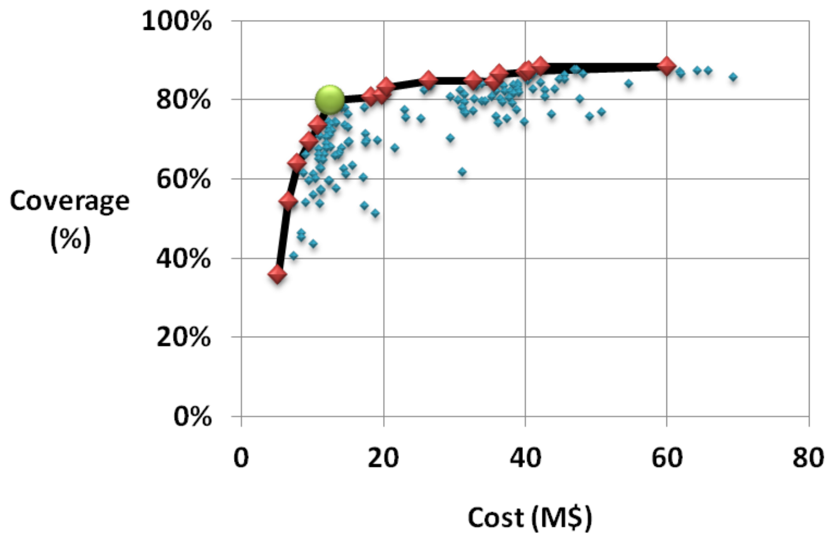


Figure 87: Fixed "Coverage-to-Cost Pareto Efficient" Detection Architecture

Table 28: Properties of the Fixed "Coverage-to-Cost Pareto Efficient" Detection Architecture

Property	Value
Total Number of Systems	124
Total Number of Radars	55
Total Number of Cameras	69
Coverage (%)	80
Cost (M$)	12.5
Number of High Cost Radars	9
Number of Medium Cost Radars	25
Number of Low Cost Radars	30
Number of High Cost Cameras	31
Number of Medium Cost Cameras	22
Number of Low Cost Cameras	7

Figure 88 shows an example simulation of the "coverage-to-cost Pareto efficient" detection architecture without the addition of mobile units. The blue persons are the CBP agents, the grey icons are the items of interest (pedestrians, cars, and trucks), the blue houses are the CBP command centers, the yellow lines model the links between detected items of interest and sensor systems that have detected them, the grey lines represent the links between sensor systems and items of interest that are within their ranges of detection but out of their lines of sight, and the red lines correspond to the links between detected items of interest and CBP agents assigned to their interception. In the simulation, it is assumed that the patrol units are traveling over a distance of 10 km, take about 5 minutes to install the detection systems on their mobile platforms, observe the surrounding environment for about 17 minutes from the closest patrol point, and take about 9 minutes to dismount the detection systems from their mobile platforms before moving on to the next patrol point. The simulation was performed for a real time period of four days. The initial conditions are summarized in Table 29.

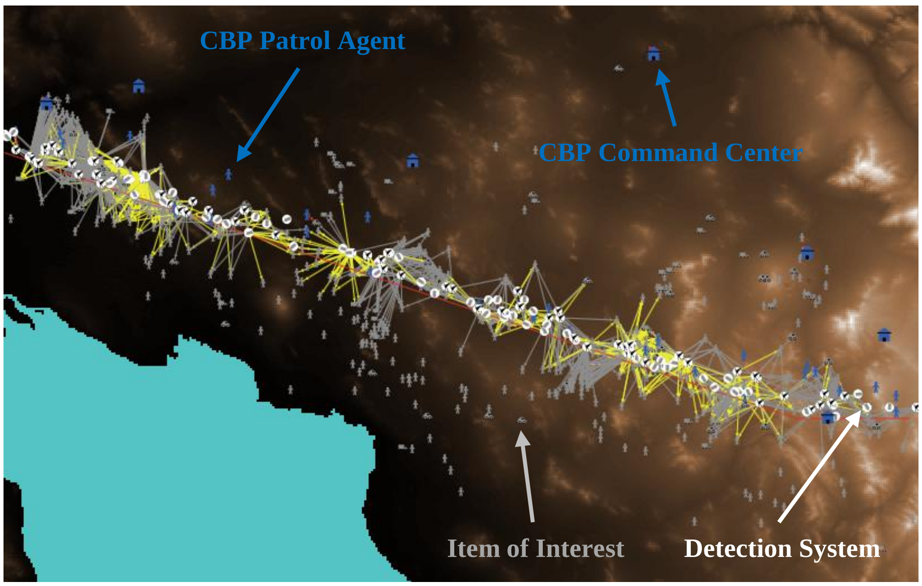


Figure 88: Simulation of the Fixed "Coverage-to-Cost Pareto Efficient" Detection Architecture, in the NetLogo Environment

Table 29: Initial Conditions for the Simulation of the Fixed Detection Architecture

Parameter	Initial Value
Number of Pedestrians Modeled	400
Number of Cars Modeled	40
Number of Trucks Modeled	30
Number of CBP Agents Modeled	30

7.2. Addition of Mobile Detection Systems to the Fixed Detection Architecture for the Customs and Border Protection Scenario

The structure of the fixed “coverage-to-cost Pareto efficient” detection architecture may be visualized in the NetLogo environment to identify regions along the border which lack detection coverage. This is depicted in Figure 89, where the regions lacking coverage are circled. In Figure 89, the black dots are the detection systems, the detection coverage of the architecture corresponds to the green areas, and the regions lacking coverage are represented by the yellow areas.

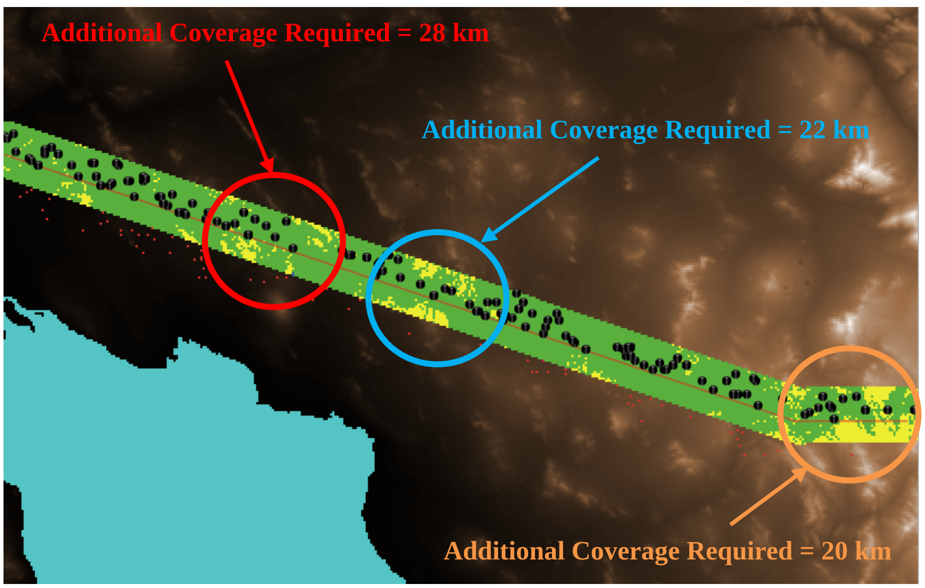


Figure 89: Visual Representation of the Detection Coverage of the Fixed “Coverage-to-Cost Pareto Efficient” Detection Architecture in the NetLogo Environment

From Figure 89, three regions requiring additional coverage may be identified. The eastern region (circled in orange) necessitates an additional coverage of about 20 km, while the western region requires an additional coverage of about 50 km that can be divided into two regions: one of 28 km radius (circled in red) and the other of 22 km radius (circled in blue). In those regions, the detection efficiency of the fixed detection architecture may be enhanced by adding mobile detection units transported by CBP agents patrolling between specifically determined patrol points. On the one hand, the additional eastern coverage can be provided by a High Cost Camera, whose design detection range is about 15 km. On the other hand, the additional detection coverage needed in the western region can be provided by a High Cost Radar and a Medium Cost Radar, whose design detection ranges are 26 km and 21 km respectively. The mobile detection units and their corresponding patrol points are pictorially represented by the blue vehicles and the blue flags respectively in Figure 90. The complete detection architecture, consisting of both fixed and mobile detection systems, is described in Table 30 and compared with the fixed detection architecture.

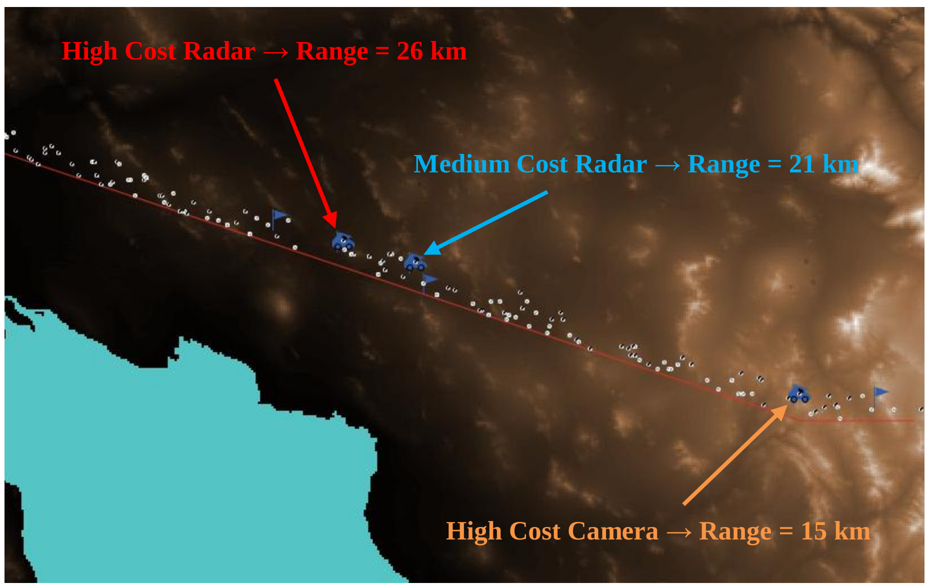


Figure 90: Visual Representation of the Complete "Coverage-to-Cost Pareto Efficient" Detection Architecture in the NetLogo Environment

Table 30: Properties of the Complete "Coverage-to-Cost Pareto Efficient" Detection Architecture Compared to the Corresponding Fixed Detection Architecture

Property	FIXED ARCHITECTURE	COMPLETE ARCHITECTURE
Total Number of Systems	124	127
Total Number of Radars	55	57
Total Number of Cameras	69	70
Coverage (%)	80	84.1
Cost (M$)	12.5	13
Number of High Cost Radars	9	10
Number of Medium Cost Radars	25	26
Number of Low Cost Radars	30	30
Number of High Cost Cameras	31	32
Number of Medium Cost Cameras	22	22
Number of Low Cost Cameras	7	7

Table 30 shows that intelligently adding mobile detection systems to the "coverage-to-cost Pareto efficient" detection architecture considered in Figure 87 and Table 28 increases the coverage by 5% while increasing the cost by 4%. The resulting complete detection architecture therefore presents a significant advantage in coverage while minimally increasing the resulting cost. Its position in the graph depicting detection architecture coverage as a function of cost, relative to the corresponding fixed "coverage-to-cost Pareto efficient" detection architecture, is represented in Figure 91 as the orange circle.

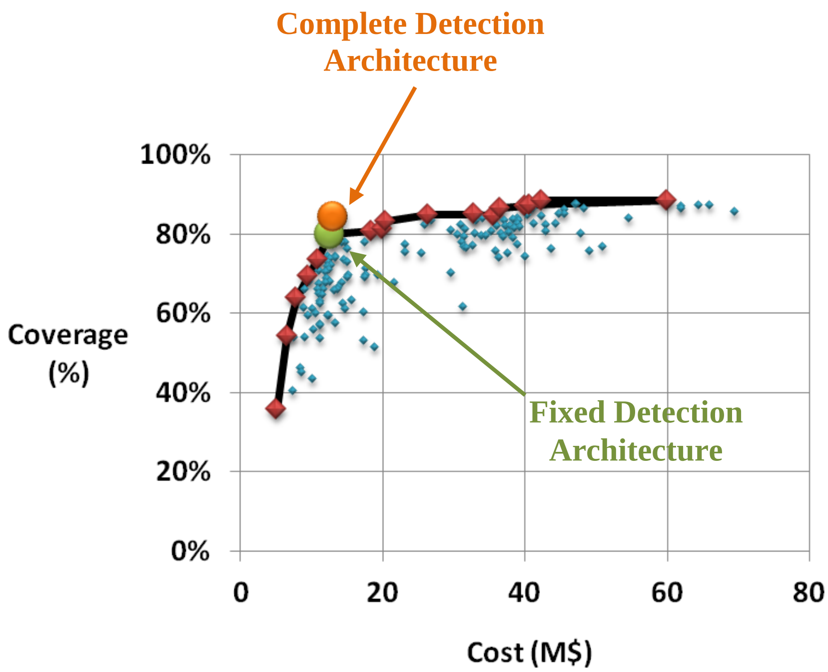


Figure 91: Complete "Coverage-to-Cost Pareto Efficient" Detection Architecture

Finally, the structure of the detection architecture composed of both fixed and mobile detection systems may be visualized in the NetLogo environment as depicted in Figure 92, alongside the structure of the corresponding fixed detection architecture (cf. Figure 89). In Figure 92, the black dots are the detection systems, and the detection coverage of the architecture corresponds to the green areas. One can readily notice that the coverage of the detection architecture is notably enhanced by the addition of mobile detection systems.

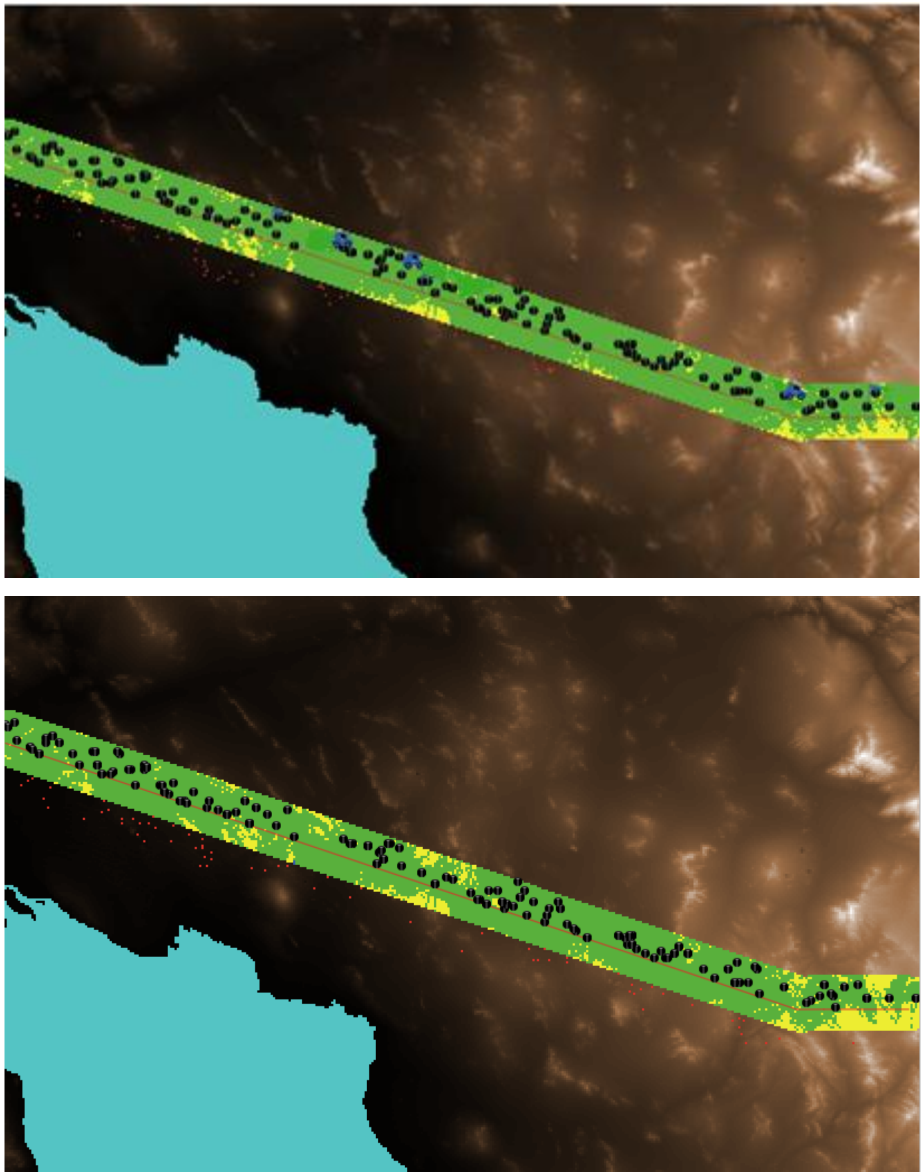


Figure 92: Visual Representation of the Detection Coverage of the Complete "Coverage-to-Cost Pareto Efficient" Detection Architecture (Top), and of the Corresponding Fixed Detection Architecture (Bottom) in the NetLogo Environment

7.3. Simulation of the Complete Detection Architecture for the Customs and Border Protection Scenario and Comparison with the Fixed Detection Architecture

Figure 93 shows an example simulation of the “coverage-to-cost Pareto efficient” detection architecture augmented with the mobile patrol units depicted as the blue cars. Again, the blue persons are the CBP agents, the grey icons are the items of interest (pedestrians, cars, and trucks), the blue houses are the CBP command centers, the yellow lines model the links between detected items of interest and sensor systems that have detected them, the grey lines represent the links between sensor systems and items of interest that are within their ranges of detection but that are out of their lines of sight, and the red lines correspond to the links between detected items of interest and CBP agents assigned to their interception. In the simulation, it is assumed that the patrol units are traveling over a distance of 10 km, take about 5 minutes to install the detection systems on their mobile platforms, observe the surrounding environment for about 17 minutes from the closest patrol point, and take about 9 minutes to dismount the detection systems from their mobile platforms before moving on to the next patrol point. The simulation was performed for a real time period of four days. The initial conditions are identical to those described in Table 29.

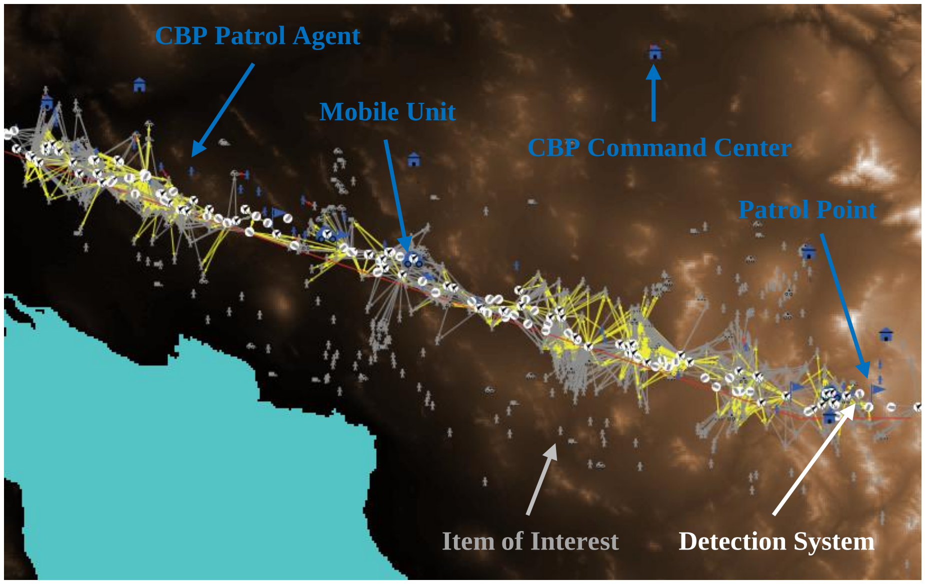


Figure 93: Simulation of the Complete "Coverage-to-Cost Pareto Efficient" Detection Architecture in the NetLogo Environment

The fixed detection architecture and the complete detection architecture with mobile systems were simulated several times in the modeling and simulation environment developed in NetLogo, under the same conditions as those described in Figure 88 and Figure 93. The resulting detection and interception performances for the fixed and the complete detection architectures were then averaged over the set of simulations performed. The average results are summarized in Table 31.

Table 31: Average Detection and Interception Performances of the Fixed and of the Corresponding Complete "Coverage-to-Cost Pareto Efficient" Detection Architectures

Average Performance Tracked During Simulation	Fixed Detection Architecture	Complete Detection Architecture
Number of Items of Interest That Have Crossed the Border	1517	1534
Number of Items of Interest That Have Been Intercepted	728	986
Number of Items of Interest That Have Escaped Detection and Interception	550	423
Number of Items of Interest That Have Been Detected	1673	1897

From Table 31, one can notice that adding mobile detection systems to the fixed "coverage-to-cost Pareto efficient" detection architecture notably enhances the average detection and interception performances. For instance, for about the same number of items of interest having crossed the border, the complete detection architecture leads to about 35% more interceptions by CBP agents on average, compared to the fixed detection architecture. The complete detection architecture also detects about 13% more items of interest on average. Finally, on average, about 23% fewer items of interest illegally escape detection by the complete detection architecture and interception by CBP agents.

Disclaimer: the previous results are generated from a sample of stochastic simulations. Therefore, one cannot draw general conclusions about the confidence in the results.

7.4. Concluding Remarks on the Simulation and Analysis of Detection Architectures for the Customs and Border Protection Mission Scenario

The previous sections have demonstrated that the flexible agent-based and physics-based framework developed in this work could be used to rapidly, quantitatively, and efficiently evaluate the operational effectiveness of a portfolio of "coverage-to-cost Pareto efficient" distributed system architectures and to identify regions on the theater of operations lacking detection capabilities. The aforementioned M&S framework could also be employed to complement a fixed detection architecture with mobile detection systems so as to increase its operational effectiveness, and to assess the relative impacts of this structural modification on the resulting detection and interception performances. Moreover, it was shown that the agent-based and physics-based framework was flexible enough for the decision maker to play "what-if" scenarios or reliability analyses, thus exploring the impact of varying the structure of the distributed system architecture on the performance and the cost metrics. The previous sections finally demonstrated the development of a quantitative, transparent, adaptive and practical methodology, while ensuring the traceability and the adequacy/validity of the definition of accurate distributed system architectures for a specific homeland security mission scenario. This step addressed the solutions analysis and "what-if" analysis research questions, and validated the corresponding hypotheses.

With this, the second leg of the "Vee" diagram is complete as depicted in Figure 94.

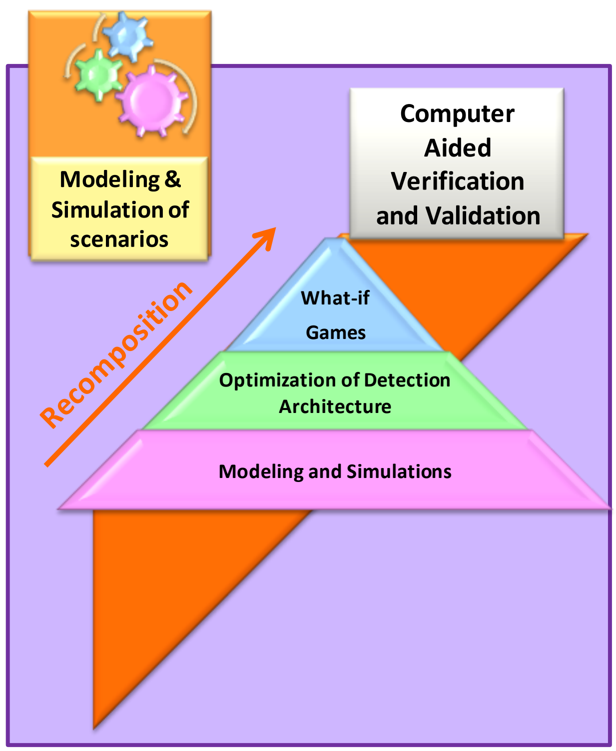


Figure 94: Recomposition of the Problem

CHAPTER VIII

CONCLUSIONS AND FUTURE WORK

Through the use of enabling methods and tools such as Functional Decomposition from Systems Engineering, Hierarchical Analysis, Morphological Analysis, Interactions and Consistency Analysis, Evolutionary Optimization, Comparative Study, the agent-based modeling and simulation tool NetLogo and the interactive optimization environment MATLAB, the objectives set for this Ph.D. research thesis were successfully met.

A seven-step methodology was proposed to facilitate the traceable, structured, and reproducible design, modeling, simulation, and optimization of distributed system detection architectures (DODA) for surveillance missions in the context of homeland security. The proposed methodology considers a set of heterogeneous detection systems and distributes them over large areas of interest in order to design detection architectures providing the maximum global detection coverage at reasonable costs in specific topographic and climatic environments. Additionally, it enables the decision maker to really understand the nature of the detection architectures, to assess their capabilities through a number of notional “what-if” scenarios, and to analyze the relative sensitivity of trade-offs at each level of the problem.

After studying the problem of DODA in many details, it is necessary to take a broad perspective and recall the main point of this dissertation. This last section steps through the five main sets of research questions and hypotheses of this thesis and summarizes the major findings associated with each set.

8.1. First Set of Research Questions and Hypotheses – Morphological Analysis

The first set of research questions concerned the ability of the decision maker to model various states of the world through a wide range of both existing and notional entities composing the homeland security mission of interest. To address this set of research questions, the general problem of DODA was largely studied and decomposed into main parameters of importance for two homeland security mission scenarios, namely the Critical Assets Protection (CAP) mission scenario and the Customs and Border Protection (CBP) mission scenario. It was shown that a structured, yet flexible, characterization scheme, built on the concept of parametric representations, provides a way to fulfill both the need to characterize a wide range of existing elements, and the incentive to generate new notional ones in a single step. In this approach, the homeland security problem of interest was progressively decomposed into its main elements, both physically and functionally. This enabled the generation of parametric representations by adequately regrouping different elements of the problem, eventually revealing sets of common parameters. It was also shown that incorporating a multi-level approach to the original Morphological Analysis (MA) method and combining it with Hierarchical Decomposition methods and with fundamental systems engineering concepts of decomposition and synthesis, such as the systems engineering “Vee” diagram, enables the creation of a rigorous, structured, and traceable analysis process. It was further demonstrated that the proposed process provides a means to determine a set of relevant alternatives that best matches all levels of decomposition, and allows accommodating any successive decomposition steps that may be required. Finally, it was possible to recompose or synthesize the problem so that lower level representations and interactions may be revealed. In this scheme, the elements of interest in the DODA problem were identified as being the defended asset (critical asset or border area), the detection system(s), the topographic environment, the items of interest that could be potential “threats” to the defended asset, and the climatic conditions under which a scenario may be taking place. Then, several alternatives were brainstormed for each parameter and regrouped in a High Level Morphological Matrix (HLMM). Subsequently, the problem was further decomposed, and relevant attributes for each element of the problem were identified. Appropriate representations of attributes (ranges of values, discrete values, and qualitative concepts) were also determined for each parameter. This allowed the creation of Sub-Level Morphological Matrices (SLMMs).

8.2. <u>Second Set of Research Questions and Hypotheses – Cross-Consistency Assessments</u>

The second set of research questions concerned the assessment of the nature of the relationships between the various elements of the problem identified from its functional and physical decomposition so as to “reduce” the number of scenarios that may be played to an operationally relevant set. To address this set of research questions, the consistency of the information contained in the Morphological Matrices was assessed and summarized in a High-Level and several Sub-Level Cross-Consistency Matrices (HLCCM and SLCCMs). This paved the way for the modeling and simulation of candidate scenarios. In this process, it was demonstrated that the traditional binary scale used to study the compatibilities between alternatives in the original morphological analysis formulation was neither sufficient nor appropriate. In order to capture complex interactions between the various elements of the problem, cross-consistency assessments based on probabilistic or likelihood representations were developed. It was shown that this probabilistic approach enables resolving the ambiguities resulting from the pairwise assessment of element compatibilities and facilitates the encoding of relational data with higher resolution scales. It was further demonstrated that such likelihood cross-consistency assessments provide a way to examine the internal relationships between the various elements of the problem and to describe the relative consistencies at each level of decomposition identified in the Morphological Matrices. It was finally noticed that they enable characterizing the potential coexistence of alternatives in notional scenarios of interest, thus facilitating the reduction of the number of scenarios that may be played to an operationally relevant set.

8.3. <u>Third Set of Research Questions and Hypotheses – Modeling and Simulation Environment</u>

The third set of research questions was twofold. First, it concerned the ability of the decision maker to determine a portfolio of architectures of heterogeneous distributed detection systems interacting with each other and with their surrounding operational, geographic and climatic environments. Then, it asked for a way to accurately, rapidly and efficiently capture the impact of changes in the operational situation on the structure (composition and design) of the distributed detection system architecture. To address this set of research questions, NetLogo and MATLAB were identified as relevant modeling, simulation, and optimization tools for providing a structured, traceable, and reproducible framework in which surveillance and detection missions may be analyzed in the context of homeland security. It was shown that physics-based models of the main elements of the problem need to be created and combined in the agent-based M&S framework NetLogo so as to obtain a portfolio of detection architectures optimized for a specific homeland security mission. In this context, it was identified that the goal is to concurrently optimize the number, the types, the properties, and the positions of heterogeneous distributed systems, so as to define detection system architectures adapted to a specific operational scenario, according to performance requirements and/or cost constraints. Finally, it was demonstrated that the M&S framework further enables the identification of key factors driving the structure of the distributed system architecture according to changes in the operational conditions.

8.4. Fourth Set of Research Questions and Hypotheses – Optimization

The fourth set of research questions concerned the selection, tuning, and testing of an optimization method to find coverage- and cost-efficient surveillance architecture solutions for the homeland security mission of interest. To address this set of research questions, two candidate evolutionary optimization approaches (genetic algorithm and particle swarm optimization) were identified to potentially solve the homeland security optimization problem. It was shown that evolutionary optimization provides a means to solve multi-objective, discontinuous and non-linear optimization problems, to balance the tradeoff between exploration and exploitation, to find a number of reliable solutions rather than a single solution for the distributed system architectures in specific operational contexts, to handle performance and/or cost constraints, and to explore the search space more thoroughly with smaller numbers of objective function evaluations. The genetic algorithm and particle swarm optimization algorithm were then adapted and refined from their original versions in order to account for the peculiar characteristics of the homeland security application.

Next, it was demonstrated that appropriate sets of algorithm parameters adapted to the homeland security mission of interest may be obtained by applying the modified optimization algorithms to simpler analytical test problems (whose solutions are known) presenting similar discontinuous, non-linear, and dimensional properties as the original problem. The optimization parameters were varied and the sensitivity of the resulting solutions to the parameter settings and combinations were analyzed. Finally, it was shown that the sets of algorithm parameter values that provided the most accurate solutions for the test problems, that ensured good convergence properties of the modified optimization algorithms and the adequacy of the resulting solutions highly depends on the maximum size of the detection architecture. Lastly, it was assumed that the resulting set of algorithm parameter values was able to ensure the convergence of the optimization algorithm to accurate distributed system architecture solutions for the homeland security application of interest.

In addition to the aforementioned sensitivity analysis, the convergence properties of the candidate evolutionary optimization algorithms were studied and compared to determine the optimization method which globally presented the best performance and the lowest computational cost. It was demonstrated that the particle swarm optimization approach presents better performance, lower computational cost and is more suitable for finding solutions to the original optimization problem than the genetic algorithm.

Subsequently, a heuristic recursive optimization scheme was developed in order to check the accuracy of the solutions provided by the modified particle swarm optimization algorithm when applied to the original homeland security application. The recursive approach was based on simple performance, cost and geometrical positioning rules. It enabled the construction of benchmark detection architectures against which the properties of the Pareto efficient detection architectures provided by the modified optimization algorithm could be compared. It was shown that the modified evolutionary algorithm is successful at finding reliable detection architectures able to satisfy the constraints of the homeland security mission scenario. In other words, the adapted particle swarm optimization method is able to efficiently optimize concurrently the number, the types, the properties, and the positions of a set of heterogeneous detection systems over a large area of operations for a specific homeland security mission, given performance requirements and/or cost constraints.

8.5. Fifth Set of Research Questions and Hypotheses – Solutions Analysis and "What-if" Analysis

The fifth set of research questions concerned the rapid, quantitative, and efficient assessment of the operational effectiveness of the portfolio of coverage- and cost-efficient distributed system architectures, and the analysis of the sensitivities of the distributed system architecture performance and cost to changes in its structure.

It was shown that the flexible agent-based and physics-based framework developed in this work allows the rapid, quantitative, and efficient evaluation of the operational effectiveness of a portfolio of Pareto efficient detection architectures obtained with the modified particle swarm optimization algorithm. The aforementioned detection architectures were initially composed of fixed detection systems distributed over the area of interest and had limited coverage performance in some identified regions of the theater of operations. Using the M&S framework created as part of this research, it was demonstrated that the fixed detection architectures may be complemented, and their global operational performance may be enhanced, by adding mobile sensor systems transported by patrolling agents on mobile platforms and deployed and operated in areas lacking coverage capabilities.

In a last step, it was shown that the aforementioned M&S environment allows the decision maker to play "what-if" scenarios, thus exploring the effects of varying the structure of the distributed system architectures on the performance and the cost metrics.

8.6. <u>Summary and Future Research Directions</u>

Ultimately, a quantitative, transparent, adaptive and practical methodology was developed to ensure the traceability and the adequacy/validity of the definition of coverage- and cost-efficient distributed system architecture solutions for a specific homeland security mission.

The results derived from the modeling, simulation, and optimization (MS&O) of detection architectures in the context of a particular CBP homeland security mission scenario helped gain insight into the characteristics of the global problem of DODA for homeland security applications, so as to effectively and efficiently detect items of interest in specific operational environments. They further enabled the identification of future tasks.

This research is primarily focused on ground items of interest. However, the MS&O environment has been developed to accommodate any other type of items of interest such as air and marine vehicles that might be relevant to other homeland security applications. Similarly, this research is based on a reduced number of detection system types that are limited to ground applications. However, the MS&O environment has the additional feature that the user can add any type of sensor systems that might be relevant to the multispectral optimization of detection architectures satisfying any kind of homeland security mission. In the same line of thought, airborne platforms may be modeled so as to provide more flexibility and mobility to the detection architectures. Such platforms are indeed able to carry around various detection systems and thus are potentially able to enhance the detection capabilities of the global detection architectures.

This research is focused on **<u>detection</u>** of items of interest near a border region. The interception aspect of the problem has been touched upon through the modeling of CBP agents behaving according to simple rules of motion. In the future, the interception of potentially suspicious items of interest may be accounted for more accurately. For instance, this may be implemented by taking into account the ability of the items of interest to learn from past experience and to adapt to the surveillance architecture.

In this context, it may also be of interest to study the dynamic reconfiguration of the detection architecture as the external operational situation evolves and as items of interest learn and diversify their behaviors. As part of the methodology itself, the analysis of the sensitivities to changes in the structures of the distributed system architectures on their performance and cost using the flexible agent-based and physics-based framework developed as part of this research has only been brushed upon and has mostly been left to the user. The investigation of such sensitivities may be the topic of a prospective reliability study to determine robust detection architectures under a variety of operational conditions.

The present work focused on the determination of detection architectures for homeland security missions in which the interest of the decision maker is equally spread between coverage performance and cost. This was implied in the choice of the weight factor α representing the relative importance of the coverage and cost metrics in the objective function used in the optimization. In this study, it was assumed that $\alpha = 1$. Nevertheless, in order to capture various decision maker preferences for a given homeland security mission, it might be of interest to perform a parametric analysis on the weight factor α, and to investigate the relative changes in the structure, properties, and operational performance of the resulting detection architectures.

In this research, the sensitivity of the performance of the candidate genetic and particle swarm optimization algorithms to the optimization parameter settings has been studied on test functions presenting similar characteristics as the original homeland security application. However, it might be interesting to perform this sensitivity analysis on the original homeland security problem in order to fine-tune even more the algorithm parameter values for applications similar to the particular CBP optimization problem considered in this research.

Finally, a "coverage-to-cost Pareto efficient" fixed detection architecture and the corresponding complete detection architecture with mobile systems were simulated several times in the modeling and simulation environment developed in NetLogo, under the same operational conditions. The resulting detection and interception performances were then averaged over the set of simulations performed for both detection architectures. In this case, no general conclusions could be drawn about the confidence in the results. It might therefore be interesting to study the confidence in the aforementioned detection and interception performances by considering a more complete set of operational scenarios and associated simulations.

8.7. Research Contributions

Last but not least, the main contributions of the present work are:

1. The development of a structured, traceable, and reproducible methodology for the design and optimization of customized detection architecture solutions in the context of homeland security applications. More precisely, the proposed methodology addresses the appropriate distribution of heterogeneous systems over large areas to provide adequate global coverage of a specific critical asset of interest at a reasonable cost. It further enables the decision maker to truly understand the nature of the distributed systems architectures, to assess their capabilities, and to capture key trade-offs between various elements of the problem by playing "what-if" scenarios in a specially developed modeling and simulation environment.
2. The incorporation of a multi-level approach to the original morphological analysis process to structure the functional and physical decomposition of the DODA problem and to provide a set of alternatives that is neither unmanageable nor incomplete. This hierarchical decomposition method enables the determination of a set of alternatives that best matches all levels of decomposition, and accommodates any successive decomposition steps that may be required, thus more closely following the conceptual formulation of the systems engineering "Vee."
3. The introduction of cross-consistency assessment methods based on probabilistic or likelihood representations to provide a way to describe the relative consistencies at each level of decomposition identified in the morphological analysis, as well as the coexistence of alternatives in various operational scenarios depending on their characteristics. Such cross-consistency assessment schemes

enables encoding relational data with higher resolution scales to capture more complex interactions, thus establishing the combinatorial logic that drives the problem synthesis into a number of internally consistent operational configurations.

4. The adaptation of the structure of chromosomes and particles as traditionally used in popular versions of genetic algorithm and particle swarm optimization to account for the peculiar characteristics of the homeland security application.
5. The definition of a rigorous methodology for the careful determination of appropriate optimization parameters for the problem under study using relevant test functions, and for the traceable down-selection of an optimization algorithm that will prove successful at solving the original optimization problem.
6. The development of a heuristic recursive optimization approach to provide benchmark solutions for the evolutionary optimization.
7. The development of a MS&O framework for subsequent "what-if" games on, and sensitivity analysis of, the Pareto optimal solutions provided by the evolutionary optimization in the context of homeland security.

With the proposed methodology and the design, modeling, simulation, and optimization (MS&O) framework developed as part of this research, decision makers in the field of homeland security now have at their disposal a means to:

- Fully understand the nature of the homeland security mission of interest
- Determine accurate solutions to the homeland security problem considered according to specific performance requirements and cost constraints
- Design new solutions adapted to specific operational situations
- Model and simulate a wide range of real-life agents and topographic, climatic, and operational situations
- Model the adaptive behavior of items of interest
- Analyze the internal relationships and compatibilities between the various agents modeled, and between the agents and the operational conditions
- Select relevant scenarios representative of the operational situation of interest
- Gain insight into an operational situation

- Assess the performance of architecture solutions to a homeland security mission in relevant operational scenarios
- Demonstrate the capabilities of systems-of-systems solutions to a homeland security mission of interest to potential customers
- Adapt existing solutions to changing operational conditions
- Enhance the operational effectiveness of existing solutions according to performance results when simulated under relevant operational scenarios
- Analyze the response of solutions to a homeland security problem to extreme operational situations
- Generate a vulnerability profile of architecture solutions to a homeland security mission
- Capture the key trade-offs at each level of decomposition of the problem
- Perform sensitivity analyzes, comparative studies, reliability analyzes, efficiency studies, etc
- Perform any kind of "what-if' analyzes on the solutions to the homeland security problem by modifying the structure of the solutions, their composition, the operational conditions, or any combinations of the above

Appendix A – More on Linear Programming (LP)

Duality Theory

Every linear program, called primal problem, can be converted into a dual problem, the solution of which provides an upper bound on the primal optimal solution [185].
For instance, if the primal problem is expressed in standard form as follows:

$$\text{maximize } x_0 = c^T x = \sum_{i=1}^{n} c_i x_i$$

$$\text{subject to } Ax \leq b \text{ or } \sum_{i=1}^{n} a_{ij} x_i \leq b_j \text{ for } j=1,\ldots,m$$

$$\text{with } x_i \geq 0 \text{ for } i=1,\ldots,n$$

Then, the symmetric dual problem can be specified as:

$$\text{minimize } y_0 = b^T y$$

$$\text{subject to } A^T y \geq c$$

$$\text{with } y_j \geq 0 \text{ for } j=1,\ldots,m$$

If the constraint $x \geq 0$ in the standard form of the primal problem is relaxed, then the corresponding dual problem can be written in an asymmetric form as:

$$\text{minimize } y_0 = b^T y$$

$$\text{subject to } A^T y = c$$

$$\text{with } y_j \geq 0 \text{ for } j=1,\ldots,m$$

Duality theory is characterized by two fundamental ideas. On the one hand, the dual of a dual linear program is the original primal linear program. On the other hand, every feasible solution of a dual linear program provides a bound on the optimal value of the primal objective function and vice versa. In this context, the weak duality theorem states that any feasible solution to the dual problem gives a dual objective function value which is always greater than or equal to the value of the primal objective function at any feasible solution. As for the strong duality theorem, it states that, if an optimal solution $x*$ to the primal problem exists, then the dual problem also has an optimal solution $y*$ defined by: $c^T x^* = b^T y^*$. Furthermore, duality theory provides a means to determine whether a linear program is unbounded or infeasible. Indeed, if the primal problem is unbounded, then the dual problem has no feasible solution by the weak duality theorem. Likewise, if the dual problem is unbounded, then the primal must not have any feasible solution [186],[187]. Nevertheless, the Farkas' lemma states that both primal and dual problems may be unsolvable and have no feasible solution [188],[189].

Covering and Packing Problems

Linear programs are also characterized by covering-packing dualities. Covering problems are minimization, usually linear, programming problems encountered in combinatorics and computer science [190]. They characterize the "covering" property of a combinatorial structure with respect to another structure, depending on its size. Examples of covering problems are the set cover problem or hitting set problem, the vertex cover problem and the edge cover problem. A covering linear program may be defined as follows:

$$\text{minimize } y_0 = b^T y$$

$$\text{subject to } A^T y \geq c$$

$$\text{with } y_j \geq 0 \text{ for } j=1,\ldots,m$$

where the matrix A has positive coefficients, and the vectors b and c have non-negative components. The dual of a covering problem is called a packing linear program, expressed in the following form:

$$\text{maximize } x_0 = c^T x = \sum_{i=1}^{n} c_i x_i$$

$$\text{subject to } Ax \leq b \text{ or } \sum_{i=1}^{n} a_{ij} x_i \leq b_j \text{ for } j=1,..,m$$

$$\text{with } x_i \geq 0 \text{ for } i=1,\ldots,n$$

where the matrix *A* has positive coefficients, and the vectors *b* and *c* have non-negative components. Packing problems are a class of maximization optimization problems encountered in mathematics. They attempt to pack identical objects together in a single two- or three-dimensional convex region or in an infinite space, as densely as possible without overlaps between objects or with the walls of the packing space. They are often related to real life packaging, storage and transportation problems, where the goal is to determine the configuration of objects which yields the maximum packing density. Examples of packing problems are the set packing problem, the matching problem, and the independent set problem, which are the respective dual problems to the set cover problem, the vertex cover problem and the edge cover problem. Other examples of packing problems include hexagonal packing of circles, packing of ellipsoids [191], three-dimensional sphere packing [192] in a face-centered cubic lattice, in a Euclidian ball, or in a cuboid, three-dimensional packing of Platonic solids [193],[194], such as cubes, tetrahedra [194],[195],[196], and octahedra, two-dimensional packing circles in a circle, a square [197],[198], an isosceles right triangle [199], or an equilateral triangle [200], two-dimensional packing squares in a square [201],[202], or a circle, and two-dimensional packing identical or different rectangles in a rectangle [203].

Covering and packing problems are important in the study of approximation algorithms [190]. Indeed, they are commonly considered linear programming relaxations, or approximations, of a pure linear program in which the variables are restricted to the two integer values 0 and 1. In this relaxation scheme, the constraint that each variable must be 0 or 1 is replaced by the weaker constraint that each variable has to belong to the interval

[0,1], namely $x_i \in \{0,1\}$ becomes $0 \leq x_i \leq 1$. The resulting linear program is solvable in polynomial time and the solution to the relaxed linear program provides clues about the solution to the original integer program.

Another property of dual problems is that an optimal solution to the dual can be obtained when an optimal solution to the primal exist using the complementary slackness theorem [204]. The theorem states that, if $\vec{x} = (x_1, x_2, \dots, x_n)$ is a feasible solution to the primal, $(w_1, w_2, \dots, w_m)$ are the corresponding primal slack variables, $\vec{y} = (y_1, y_2, \dots, y_n)$ is a feasible solution to the dual, and $(z_1, z_2, \dots, z_n)$ are the corresponding dual slack variables, then $\vec{x}$ and $\vec{y}$ are optimal solutions to their respective problems if and only if $x_j z_j = 0 \; for \; j = 1,2, \dots, n$ and $w_i y_i = 0 \; for \; i = 1,2, \dots, m$. In other words, if the *i*-th slack variable of the primal is different from zero, then the *i*-th variable of the dual has to be equal to zero. Similarly, if the *j*-th slack variable of the dual is different from zero, then the *j*-th variable of the primal has to be equal to zero. Economically, this means that if there are some leftovers in a constrained primal resource, then any additional quantities of that resource must have a zero value. Similarly, if some additional quantities of a constrained primal resource have a non-zero value, then there must not be some leftovers in the primal supplies.

Solutions to Linear Programming Problems

The linear constraints in a linear programming problem define the convex polyhedron space over which the linear objective function is defined. Since a linear function is both a convex and a concave function, every local minimum is also a global minimum for minimization problems, and every local maximum is also a global maximum for maximization problems [183]. However, an optimal solution to a linear programming problem may not exist, for example when two or more constraints are contradictory or inconsistent, or when the convex polytope is unbounded in the direction of the gradient of the objective function. In this case, the linear optimization problem is said to be infeasible. When the constraints are consistent, the linear objective function is bounded, and a feasible solution exists, the optimal solution is always located at a vertex of the convex polytope [205],[206]. The vertices of the convex space are called basic feasible solutions and are at the root of the simplex algorithm for solving liner programs. Some linear programming problems may have multiple optimal solutions, for example when the objective function is the zero function. In this case, the linear optimization problem has two optimal solutions and any convex combination of these two solutions is also a solution.

Methods for Solving Linear Programming Problems

The linear programming problem was first shown to be solvable in polynomial time by Khachivan in 1979 [207]. In this case, the algorithm "rapidly" converges after a number of steps of $O(n^k)$ for a given input and some non-negative integer k, where n is the complexity of the input. In 1984, Karmarkar introduced an interior-point method for solving linear programming problems which turned out to be more practical than polynomial-time algorithms and which revolutionized the field of linear programming [208]. Basic ideas from linear programming have also inspired most of the central concepts of optimization theory, mainly the notions of duality, convexity, and problem decomposition. There exist a wide range of algorithms for solving linear optimization problems, including, but not limited to, the following:

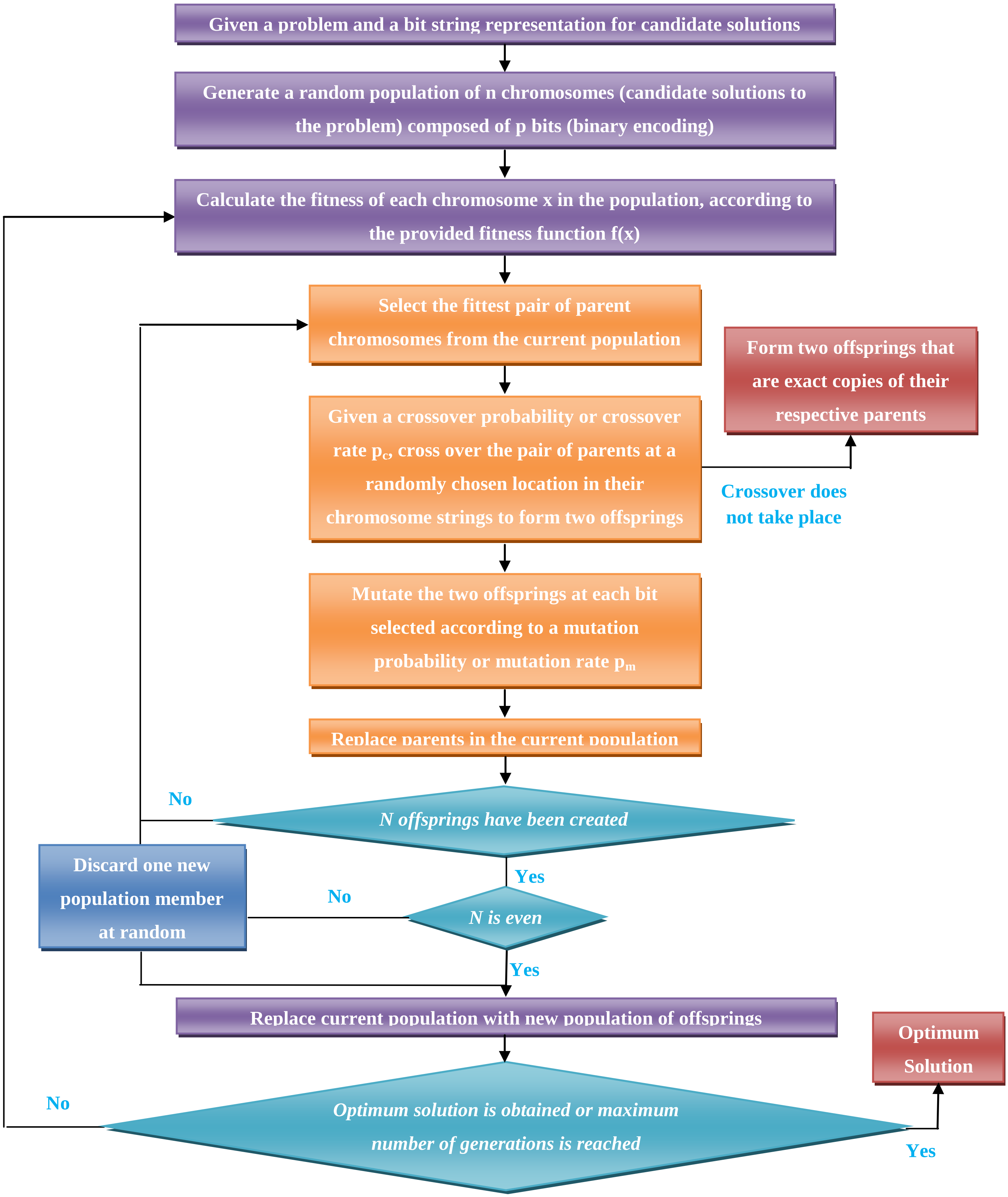
- Basic exchange algorithms
  - Dantzig's Simplex algorithm [183],[209],[210],[211]
  - Serang's Conic sampling algorithm [212]
  - Criss-cross algorithm [213],[214],[215][216]
- Interior-point algorithms
  - Khachiyan's Ellipsoid algorithm [217],[218]
  - Karmarkar's Projective algorithm [219]
  - Path-following algorithms [217],[183]

In the case where *x* is a vector of integers, the problem is an integer linear programming problem (ILP). If only some components of the unknown vector *x* are integers, the problem is a mixed integer programming problem (MILP) [220],[221]. Contrary to linear programs which can be solved efficiently, commonly encountered integer and mixed integer linear programs are often NP-hard [222]. For instance, 0-1 pure integer programming or binary integer programming belongs to this category of Karp's 21 NP-complete problems [223]. Nevertheless, there exist some special categories of ILP and MILP problems that can be solved efficiently. It is the case for problems satisfying the total dual integrality property [224], which states that, for a linear system $\mathrm{A}x \leq b$ where A and b are rational, if there exists a feasible, bounded solution to the standard linear maximization problem for any vector c, then there exists an integer optimal solution to the dual problem. Some specialized algorithms used to solve the above class of IP and MILP problems include, but are not limited to:

- Cutting-plane method [225]
- Branch and bound method [226]
- Branch and cut method [227]
- Branch and price method [228],[229]

Appendix B – More on Genetic Algorithm (GA)

Simple Genetic Algorithm



Reproduction Methods

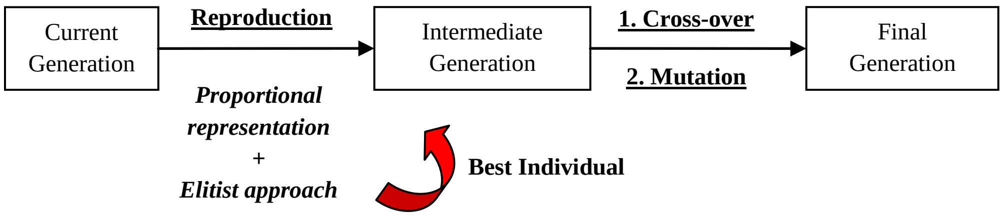
In its basic form, the "**roulette wheel**" is one of the most common techniques used for proportionate selection. The selection procedure is as follows [267]:

1. Sum the fitness of all the population members to get the total fitness F_{sum}
2. Generate a random number n between 0 and the total fitness F_{sum}
3. Return the first population member whose fitness, added to the fitness of the preceding population member, is greater than or equal to n

The "roulette wheel" usually has some problems when the fitness functions of individuals are very different from one another. For instance, if the fitness of a chromosome occupies 90% of the area of the roulette wheel, then the other chromosomes have very few chances of being selected.

The "**rank selection**" method is a modification of the "roulette wheel" method. It first ranks the population of chromosomes, and then allocates a fitness value to every chromosome from this ranking. In this scheme, the worst chromosome has fitness 1, the second worst has fitness 2, and so on, such that the best chromosome has fitness N, which is the number of chromosomes in the population. After this modification of the roulette wheel, the least fit chromosomes have a lower chance of being selected. Nevertheless, the "rank selection" method can lead to slower convergence due to the fact that, after ranking of the population, the fittest chromosome is not so much different from the other ones.

The basic "**proportional representation**" method of selection of individuals for reproduction can be modified so that the best design is always passed to the next generation. In this scheme, the best design of the current generation is passed to the intermediate generation before crossover and mutation, and can thus be subject to changes due to crossover and mutation. This is called the "elitist approach." The process used in the modified "proportional representation" with "**elitist approach**" is the following:



The best individual is therefore passed to the intermediate generation before crossover and mutation, and not after, which would be another way to implement the elitist approach.

In the "**tournament selection**" method, the chromosomes in the population are ranked according to their fitness such that the first individual of the intermediate population is the "best" member of the current generation (to ensure the compliance with the "elitist" approach). Then, two individuals are randomly selected in the current population. Among these two individuals, the one that corresponds to the highest fitness is passed to the intermediate generation. This process is repeated until a complete intermediate population is obtained.

Properties of Genetic Algorithms

Genetic algorithms have proved to be successful in a wide range of applications, especially in cases of optimization. However, one of the major difficulties in practice is the premature convergence of the algorithm which converges before the expected evolutionary target is reached. During the evolution, the genetic features of an elitist individual in the population may dominate the whole population and make the evolution stuck in a local optimum due to some loss of diversity in the population. Various solutions can nevertheless be brought forth [271], such as centralized methods (Incest Prevention and Pygmy Algorithm [272]), and distributed methods (Island Model [272]). For instance, the Genetic Algorithm with Varying Population Size or GAVaPS is a hybrid algorithm which exploits the merits of both of the two classes of techniques described above [273].

The size of the populations can also be critical in many applications of genetic algorithms due to its strong influence on the computational time. If the population size is too small, the genetic algorithm may converge too quickly. However, if it is too large, the genetic algorithm may waste computational resources: the waiting time for an improvement might be too long, and convergence of the algorithm may be strongly delayed. A solution to this problem might be to use an adaptive method for maintaining a variable population size, which grows and shrinks together according to some characteristics of the search [273].

Finally, traditional GAs have been enhanced and classified according to their properties and their applications. Some classes include [272]:

- Simple Genetic Algorithm (SGA)
- Parallel and Distributed Genetic Algorithm (PGA and DGA) such as Master-Slave Parallelization, Fine Grained Parallel GAs (cellular GAs), Multiple-Deme Parallel GAs (Distributed GAs or Coarse Grained GAs), and Hierarchical Parallel Algorithms
- Hybrid Genetic Algorithm (HGA)
- Adaptive Genetic Algorithm (AGA)
- Fast Messy Genetic Algorithm (FmGA)
- Independent Sampling Genetic Algorithm (ISGA)
- Hybrid GAs (some will be described in subsequent sections)

Appendix C – More on Simulated Annealing (SA)

Simulated Annealing works by analogy with the annealing process used in metallurgy:

1. A material is heated until its atoms are freed from their initial configuration corresponding to a local minimum of its internal energy.
2. The free atoms can then wander randomly through states of higher energy before being slowly cooled down.
3. Controlled cooling of the material increases the chance of freezing its atoms in a configuration of lower internal energy than the initial one.

Based on the previous concepts, the SA approach to combinatorial problems is straightforward:

1. The current thermodynamic state of the system is equivalent to the current solution to the combinatorial problem.
2. The energy equation for the thermodynamic system is similar to the objective function of the combinatorial problem.
3. The ground state of the thermodynamic system is analogous to the global minimum of the objective function.

Similarly to the metallurgic process, SA features a parameter T, called temperature that is gradually decreased during the optimization to simulate the cooling process. At each step of SA, the temperature parameter is used to replace the current solution by a random nearby solution, chosen according to a probability dependent on the difference between the corresponding function values. In this scheme, the current solution changes somehow randomly at the beginning, when T is large, but increasingly "downhill" as the optimization progresses, i.e. as T decreases to zero. In other words, as cooling proceeds, the material becomes more and more ordered, and approaches a frozen ground state at T = 0. Nevertheless, the probabilistic allowance for "uphill" moves saves the algorithm from being stuck at local optima: the process of SA can be thought of as an adiabatic approach to the lowest energy state, such that if the initial temperature is too low or if the cooling is done not slowly enough, the material may freeze in a metastable state, i.e. may be trapped in a local minimum energy state [277]. However, the major difficulty in implementing SA resides in the fact that the temperature T of the thermodynamic system cannot be represented as a free parameter in the combinatorial problem. Besides, the ability of the SA algorithm to not be stuck at a local minimum depends on the *"annealing schedule"* [277], the choice of the initial temperature T, the number of iterations performed at each temperature, and the rate at which the temperature is decreased as the cooling process progresses.

In short, given a neighborhood structure, the SA algorithm features a procedure that continuously attempts to transform the current configuration into one of its neighbors as follows [277]:

1. Initialization: an initial "state" is randomly selected in the search space, and the temperature T is set to a very high value (or to infinity)
2. Motion: the current "state" is perturbed according to a defined move (or transition) depending on a transition probability
3. Evaluation: the change ΔC in the objective (or cost, or energy) function between the current "state" and the new "state" is calculated
4. Choice: the previous "transition" is accepted or rejected, depending on the change in the objective function: if ΔC is negative, the move is accepted, otherwise, it is accepted according to a probability given by the Boltzmann factor $e^{-\frac{\Delta C}{T}}$, called the acceptance probability
5. Update: the temperature is updated according to an *"annealing schedule"* (cooling schedule) which gives the rate of decrease of the temperature with time (or with the number of iterations)
6. Repeat: starting back at step 2, the process is repeated until a *"freezing point,"* or a specific time allotted for computations, is reached

Appendix D – More on Tabu Search

Tabu search is composed of three steps [257],[312]:

- A preliminary search in which all the neighbors to the current potential solution *x* are evaluated and the one with the highest fitness is kept as the new potential solution *x'*. This does not imply that *x'* is better than *x*, but rather that it is the best of all the neighbors of *x*. Therefore, this step does not necessarily lead to an improved solution. However, this enables the algorithm to escape an eventual local optimum and to continue the search even when a move does not improve the solution. In such a case, the neighbor that least degrades the objective function is the one that is chosen to proceed with the optimization. In order to avoid going back and forth neighboring solutions *x* and *x'*, tabu search uses the memory structures discussed above to avoid re-visiting solutions recently encountered, over a specific number of subsequent moves. Memory structures thus work line circular lists which are empty when they are created and which regenerate based on a first-in-first-out procedure. However, because tabu lists may ban interesting moves leading to a better solution than the best one found so far, an aspiration criterion is used to allow such moves.
- An intensification phase which starts with the best solution encountered so far, then clears the tabu list, and finally applies the same steps as in the preliminary search for a given number of iterations. This phase enables the optimization algorithm to intensify the search in the most promising regions of the design space discovered during the preliminary search.
- A diversification phase in which the tabu list is cleared, a certain number of most frequent moves are set to be tabu, a random solution *x* is chosen, and the same steps as in the preliminary search are applied for a given number of moves. This phase forces the optimization algorithm to explore regions of the search space that have not been visited enough.

The description of tabu search above shows that a certain number of parameters are of potentially critical importance to the efficiency of the algorithm. These include the sizes of the neighborhoods, the types of moves, the characteristics of the memory structures, the aspiration criteria, and the definition of a variety of inputs such as the maximum number of moves, the number of iterations chosen, etc. Fouskakis [313] summarizes findings on how to make decisions about the above parameters.

Appendix E – More on Ant Colony Optimization (ACO)

Ant Colony Optimization Algorithm

The ACO algorithm works as follow [328],[329]:

1. The first ant leaves the colony in search for food, following a more or less random path until it finds a food source. The ant then returns more or less directly to the nest, laying down a trail of pheromones behind it.
2. The nearby ants are attracted by the pheromones and are likely to follow more or less directly the path found by the first ant. Upon returning to the colony, these ants also leave a trail of pheromones along the path, strengthening the initial route to the food source.
3. Over time, the pheromone trail tends to evaporate which reduces the attractive strength of the route. In this context, a short path gets explored more frequently than a long path, and thus the pheromone density increases more rapidly on shorter paths than on longer ones.
4. As a consequence, if any two different routes end up reaching the same source of food, then ants tend to use the shorter one more often than the longer one. As such, the pheromone signature of the shortest path is increasingly enhanced so that the short route becomes more and more attractive to the ants, while the pheromones along the longest path evaporates over time and the long route becomes less and less attractive.
5. Eventually, the ants figure out and choose the shortest route to a given source of food.

The original ACO algorithm involves the following main principles [337],[339]:

- A colony of ants moves through the various states of a problem according to local decision rules based on trail attractiveness. Through this process, each ant incrementally constructs a solution to the problem. Intermediate solutions are called solution states. At each step of the algorithm, each ant computes a set of feasible moves leading to a more complete solution. The probability of an ant to move from its current state to a feasible state is a function of the attractiveness of the move and of the trail level. The attractiveness of the move is a heuristic giving the a priori desirability of that particular move, while the trail level indicates the a posteriori desirability of that move.

- As an ant constructs or completes a solution, it evaluates the solution and modifies the trail value by depositing pheromones which are used to direct the search of the future ants. Trail levels are updated when all ants have completed their solution: they increase if the corresponding moves lead to a “good” solution, and decrease in the opposite case.
- As the pheromones spread along a route evaporate over time, the trail value decreases. This prevents the algorithm from getting stuck at local optima.
- The algorithm can also be biased by “daemon actions” to enhance the exploration capability of the ant colony and to avoid constraining the search to local regions.

In order to find the shortest route to a food source, ants communicate with each other by depositing pheromones that store the memory of their search and detail the status of their behavior. However, the reach of pheromones is limited and only those ants located near the pheromone trail benefit from the information it contains. This mechanism was termed “Stigmergy” by the French biologist Pierre-Paul Grassé in 1939 [331] as he was studying the behavior of termites. He defined the word as a “Stimulation of workers by the performance they have achieved.” Its meaning captures the self-organization mechanism through which agents and actions are indirectly coordinated as they leave signs in the environment [332]. The process lies in the fact that any trace left in the environment by the action of a given agent influences the next action performed by either the same or a different agent. Successive actions therefore build on each other and support the emergence of efficient collaboration, coherent activity and complex intelligent structures of simple agents lacking memory, intelligence and awareness of each other, without the need for any direct communication or control between them [333],[334]. Complex problems are typical examples of self-organized systems which are based on [335],[336]:

- A positive feedback where ants deposit pheromones along the paths they explore and attract other ants that strengthen the path by spreading pheromones themselves
- A negative feedback where the pheromones evaporate over time to prevent the system from getting stuck in suboptimal regions and to amplify short routes with strong pheromone signatures

Variations of the Ant Colony Optimization Algorithm and Applications

The first ACO algorithm, called the Ant system [338], was developed to solve the travelling salesman problem, where the goal is to find the shortest round-trip to near-optimally visit a set of cities only once. In this context, each ant moving in the search space represents a solution to the problem. Ants consider previous knowledge about paths explored by other ants to construct their solution and mark the best possible paths. ACO algorithms have then extended [339] and have been applied to a wide variety of combinatorial optimization problems [340], such as scheduling problems [341], vehicle routing problems [342], assignment problems, set problems, network problems, stochastic, continuous, and mixed-variable optimization problems [343], multi-target problems, and a lot of other dynamic problems in real variables. Many papers reporting on current research on ACO algorithms can be found in the proceedings of the ANTS conference [344] or in the Swarm Intelligence journal [345]. Some common variations of ACO include, but are not limited to, the Elitist Ant System [346],[347], the Max-Min Ant System (MMAS) [348], the Ant Colony System [349], the Rank-Based Ant System (ASrank), the Continuous Orthogonal Ant Colony (COAC) [350], and the Ant Colony Optimization with Fuzzy Logic [351].

Appendix F – More on Particle Swarm Optimization (PSO)

In the procedure of the Particle Swarm Optimization algorithm described in the main text, the particle's velocity is updated according to the global model: it takes into account the experience of the particle itself through the particle best, but also the experience of the whole swarm, including the particle, through the global best. There exist three subsets of this model:

1. The <u>cognition-only model</u> which updates the velocity of the particle according to the particle's best only.
2. The <u>social-only model</u> which updates the velocity of the particle according to the global best only.
3. The <u>selfless model</u> which is identical to the social-only model, with the exception that the performance of the swarm does not contain the individual's own previous performance.

Incidentally, Kennedy showed that the social-only model performs better than the selfless model, which performs better than the full model, which performs better than the cognition-only model [369]. There also exist several alternatives to the global model of PSO as initially created by Eberhart and Kennedy. For instance, in the local model, the particles receive information of their own best and their nearest neighbors' bests only, rather than that of the entire group [361]. However, the neighborhood of a given particle can be composed of a variable number of individuals. Eberhart and Kennedy showed that for a neighborhood of two individuals, this version of the PSO is quasi invulnerable to local optima. This might results from the fact that a number of "groups" of particles spontaneously separate and explore different regions. As such, the local model seems to be a more flexible approach than the global best model. Another example is the fully informed PSO, developed by Mendes [370]. In the fully informed PSO, a particle is attracted by every other particle in its neighborhood, i.e. is influenced by its neighbors' best positions, but not by the global best. Some other variants of PSO include PSO with dynamic neighborhood topologies [371], [372], PSO with enhanced diversity [373], [374], PSO with different velocity update rules [375], [376], PSO with components from other evolutionary approaches [367], [377], PSO for discrete optimization problems [378], [379], and so on.

Appendix G – More on Pareto Optimization

Definition of Pareto Optimality

Let *n* be the number of design parameters in a given design space, and *m* be the number of criteria used to assess each of the design points. Let *f* be the function which assigns a criteria vector to each point in the design space, thus providing a way to value each design point. Some designs may be infeasible. Let *X* be the compact set of feasible designs, and *Y* be the set of feasible criterion points, image of *X* under the action of *f*. In engineering problems, it is often assumed that the criteria used to assess each design option have to be minimized. Then, each dimension of the criterion vector *Y* has to be minimized as well. In this context, the Pareto frontier is defined as a subset of *Y* of the feasible criterion points which are not strictly dominated. In other words, a criterion vector *y* is preferred to another vector *y** if at least one parameter of *y* is strictly less than the corresponding parameter of *y**, and none other parameter of *y* is strictly greater. This is denoted as $y_i \leq y_i^*$ for each *i* and $y_i < y_i^*$ for some *i*. $y \leq y^*$ means that *y** is strictly dominated by *y*. Then, the Pareto frontier is the ensemble of points *y* in *Y* which strictly dominate any other point in *Y*. Assume that the criteria vector is composed of two objective functions f_1 and f_2, then Figure 95 illustrates the Pareto optimal frontier as a solid line in the case where the two criteria need to be minimized.

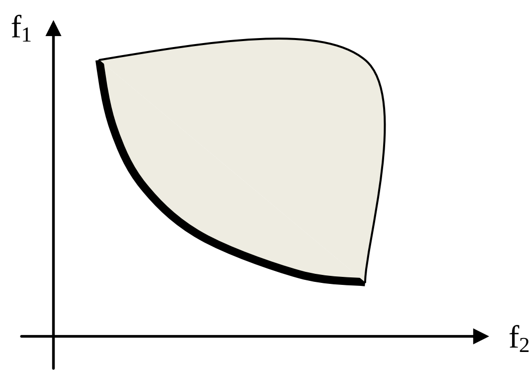


Figure 95: Illustration of a Pareto Frontier for a Two-Criteria Minimization Problem

Several algorithms have been developed to compute the Pareto frontier of a finite number of alternatives in maximum vector or skyline query problems, such as those described in Kung et al. [401] and in Parke et al. [402]

Pareto-Based Multi-Objective Optimization Algorithms

Several modified and hybrid algorithms have been proposed to overcome some of the shortcomings of traditional evolutionary algorithms, such as GA and PSO, or to combine their strengths, so as to efficiently obtain Pareto optimal solutions. Some of the most common enhanced algorithms are the following.

- Interactive Particle Swarm Optimization [365]
- Unified Particle Swarm Optimization (UPSO) [410],[411].
- Multiple objective PSO (MOPSO) [412]
- Pareto Archived Evolutionary Strategy (PAES) [413]
- Strength Pareto Evolutionary Algorithm (SPEA) and SPEA2 [414],[415], [416]
- Non-dominated Sorting Genetic Algorithm I and II (NSGA-I and NSGA-II) [417]
- Genetical Swarm Optimization (GSO) [418]
- Clonal Selection Algorithm (CLONALG) [419],[261],[420],[421], [422]
- Hybrid Algorithm of Clonal Selection Principle (CLONALG) and Particle Swarm Intelligence [423]
- Hybrid Genetic Algorithm and Particle Swarm Optimization (HGAPSO) [424]
- Comparison of Multi-Objective GAs (MOGAs)

Table 32 highlights some of the most well-known multi-objective genetic algorithms, and compares their major advantages and disadvantages.

Table 32 is directly extracted from the work of Konak et al. on MOGAs [425].

In Table 32, the following acronyms are used:

- VEGA: Vector Evaluated Genetic Algorithm
- MOGA: Multi-Objective Genetic Algorithm
- WBGA: Weight-Based Genetic Algorithm

- NPGA: Niched Pareto Genetic Algorithm
- RWGA: Random Weight Genetic Algorithm
- PESA: Pareto Envelop Selection Algorithm
- PAES: Pareto Archived Evolution Strategy
- NSGA: Non-dominated Sorting Genetic Algorithm
- SPEA: Strength Pareto Evolutionary Algorithm
- RDGA: Rank-Density Genetic Algorithm
- DMOEA: Dynamic Multi-Objective Evolutionary Algorithm

Table 32: Comparison of Multi-Objective Genetic Algorithms

Algorithm	Fitness Assignment	Diversity Mechanism	Elitism	External Population	Advantages	Disadvantages
VEGA	Each subpopulation is evaluated with respect to a different objective	No	No	No	First MOGA developed Straightforward implementation	Tends to converge to the extreme of each objective
MOGA	Pareto ranking	Fitness sharing by niching	No	No	Simple extension of single objective GA	Usually slow convergence Problems related to niche size parameter
WBGA	Weighted average of normalized objectives	Niching Predefined weights	No	No	Simple extension of single objective GA	Difficulties in non-convex objective function space
NPGA	No fitness assignment, tournament selection	Niche count as tie-breaker in tournament selection	No	No	Very simple selection process with tournament selection	Problems related to niche size parameter Extra parameter for tournament selection
RWGA	Weighted average of normalized objectives	Randomly assigned weights	Yes	Yes	Efficient and easy to implement	Difficulties in non-convex objective function space

Table 32: Comparison of Multi-Objective Genetic Algorithms (Continued)

PESA	No fitness assignment	Cell-based density	Pure elitism	Yes	Easy to implement Computationally efficient	Performance depends on cell sizes Prior information needed about objective space
PAES	Pareto dominance is used to replace a parent if offspring dominates	Cell-based density as tie-breaker between offspring and parent	Yes	Yes	Random mutation hill-climbing strategy Easy to implement	Not a population-based approach Performance depends on cell sizes
NSGA	Ranking based on non-domination sorting	Fitness sharing by niching	No	No	Fast convergence	Problems related to niche size parameter
NSGA-II	Ranking based on non-domination sorting	Crowding distance	Yes	No	Single parameter Well tested Efficient	Crowding distance works in objective space only
SPEA	Ranking based on the external archive of non-dominated solutions	Clustering to truncate external population	Yes	Yes	Well tested No parameter for clustering	Complex clustering algorithm
SPEA-2	Strength of dominators	Density based on the k-th nearest neighbor	Yes	Yes	Improved SPEA Makes sure extreme points are preserved	Computationally expensive fitness and density calculations

Table 32: Comparison of Multi-Objective Genetic Algorithms (Continued)

RDGA	The problem is reduced to a bi-objective problem, with solution rank and density as objectives	Forbidden region cell-based density	Yes	Yes	Dynamic cell update Robust with respect to the number of objectives	More difficult to implement than other MOGAs
DMOEA	Cell-based ranking	Adaptive cell-based density	Yes - implicitly	No	Includes efficient techniques to update cell densities Adaptive approaches to GA parameters	More difficult to implement than other MOGAs

Appendix H – Agent-Based Modeling and Simulation Tools and Optimization Frameworks

SWARM

The Swarm project was developed in 1994 by Chris Langton, then at the Santa Fe Institute (SFI) in New Mexico [445] and is currently based at the non-for-profit organization, Swarm Development Group also based in Santa Fe, New Mexico. Swarm is a platform for agent-based models (ABMs) and a collection of software libraries which provide support for simulation programming. The libraries are written either in objective C or in Java. The main features of Swarm are the following [446]:

- Modeling software that:
    - Is accessible
    - Can accommodate arbitrarily large experiments (given the computing resources)
    - Integrates well with other modeling techniques and technologies
    - Helps reducing error
- Agent-based modeling language that:
    - Allows the modeler to describe, combine and reuse compartments of behavior
    - Has precise and exploitable concurrency semantics
    - Is based on a few simple but flexible building blocks (messages, objects, groups and schedules)
- Support tools:
    - Space library with visualization
    - Analysis tools, graphing, etc
    - Random number generators and distributors
    - Library of GUI widgets
    - Portability and Integration:
    - Models can be written using Objective C or Java
    - Zero dependence on proprietary infrastructure

Moreover, Swarm was designed to help researchers build models of complex systems in which low-level actors interact, and to discern overall patterns that emerge from detailed behaviors at the individual level. The main drawback is that learning to program with Swarm requires reading example applications, studying the technical reference material and sometimes even getting down to the level of reviewing the header files in the libraries. It also currently requires a good hands-on knowledge of object-oriented programming and software development processes in general.

Furthermore, *"Swarm is not yet a 'shrink-wrapped' simulation toolkit. There are many of those kinds of products on the market. However, with these packages, the ease of use comes at a price – you are locked into that vendor's particular modeling paradigm. Swarm was intended to embrace many different types of modeling - consequently, it can be more difficult for a novice user – but more powerful in the long-run"* [447].

RePast (Recursive Porous Agent Simulation Toolkit)

Repast is a free and open source agent-based modeling toolkit originally developed by Sallach, Collier, Howe, North and others [448]. It was created at the University of Chicago and has subsequently been maintained by organizations such as the Argonne National Laboratory. Repast borrows many concepts from the Swarm agent-based modeling toolkit. However, contrary to Swarm, Repast has multiple implementations in several languages and has built-in adaptive features such as genetic algorithms and regression.

Besides, Repast is meant to support the development of extremely flexible models of living social agents, but is not limited to modeling living social entities alone.

A list of the main features of the last version of Repast (Repast 3), taken from the Repast website [449], is provided below:

- Variety of agent templates and examples, but the users still have complete flexibility as to how they specify the properties and behaviors of agents
- Fully object-oriented
- Fully concurrent discrete event scheduler supporting both sequential and parallel discrete event operations

- Built-in simulation results logging and graphing tools
- Automated Monte Carlo simulation framework
- Variety of two-dimensional agent environments and visualizations
- Dynamical access and modification of agent properties, agent behavioral equations, and model properties at run time
- Libraries for genetic algorithms, neural networks, random number generation, and specialized mathematics
- Built-in systems dynamics modeling
- Social network modeling support tools
- Integrated geographical information systems (GIS) support
- Repast models can be developed in many languages including Java, C#, Managed C++, Visual Basic.Net, Managed Lisp, Managed Prolog, and Python scripting

Despite and because of its numerous interesting features, Repast is a lot more difficult to learn than for instance NetLogo.

MASON

"MASON is a single-process discrete-event simulation core and visualization library aimed at multi-agent simulations with large numbers of agents" [450]. It was developed as a joint effort between George Mason University Evolutionary Computation Laboratory (ECLab) and the GMU Center for Social complexity. It stands for **M**ulti-**A**gent **S**imulator **O**f **N**eighborhoods... or **N**etworks.

"MASON was designed to be the foundation for large custom-purpose Java simulations, and also to provide more than enough functionality for many lightweight simulation needs" [451]. However, the system is written in pure Java and is intended for experienced Java coders who want something general and easily manageable to start with, rather than a domain-specific simulation environment. Although coding in Java is not a problem for the author, the learning curve of the tool is too steep. As such, MASON is not really suitable for the problem under consideration.

Nonetheless, a list of the main features of MASON, taken from the Center of Social Complexity GMU website [451], is provided below:

- 100% Java
- Fast, portable and fairly small
- Models are completely independent from visualization. They can be added, removed, or changed at any time
- Models may be check-pointed and recovered but also dynamically migrated across platforms
- Can produce reproducible results regardless of the platform
- Models are self-contained and can run inside other Java frameworks and applications
- 2D and 3D visualization
- Can generate PNG snapshots, Quicktime movies, charts and graphs, and output data streams

NetLogo

"NetLogo is a cross-platform multi-agent programmable modeling environment" developed in 1999 by Uri Wilensky in order to simulate natural and social phenomena [452]. Since then, it has been successfully and widely used for the modeling of complex systems. As for the name of the tool, the "Logo" part is because NetLogo is a dialect of the Logo language, and the "Net" part is meant to evoke the decentralized, interconnected nature of the phenomena that can be modeled with NetLogo, including network phenomena.

What is interesting with NetLogo is its agent-based nature making possible the exploration of the connection between individual behaviors of hundreds or thousands of agents operating independently, and of the world patterns that emerge from their interaction. NetLogo is not only simple enough so that models can be rapidly built and simulated by novices, but also *"advanced enough to serve as a powerful tool for researchers in many fields"* [453]. It is also publicly available. NetLogo comes with an extensive documentation and with tutorials as well as with a collection of example models and simulations that can be used and modified and that really facilitate the learning of the tool. Last but not least, *"NetLogo is the next generation of the series of multi-agent modeling languages that started with StarLogo"* [453]. As such, it adds significant new features to the functionality of StartLogoT, such as a redesigned language and user interface facilitating the ease of use for the untrained person. NetLogo is written in Java which makes it suitable for all major platforms (Mac, Windows, Linux, et al) and is in continuous development at the Center for Connected Learning and Computer-Based Modeling.

A list of the main features of NetLogo, taken directly from the NetLogo website [453], is provided below:

- Language:
    - Fully programmable
    - Simple language structure
    - Language is Logo dialect extended to support agents
    - Mobile agents (turtles) move over a grid of stationary agents (patches)
    - Create links between turtles to make aggregates, networks, and graphs
    - Large vocabulary of built-in language primitives
    - Double precision floating point math
    - Runs are exactly reproducible across platforms
- Environment:
    - View model in either 2D or 3D
    - Scalable and rotatable vector shapes
    - Command center for on-the-fly interaction
    - Interface builder with buttons, sliders, switches, choosers, monitors, text boxes, notes, output area
    - Speed slider lets fast forward the model or see it in slow motion
    - Powerful and flexible plotting system
    - Info tab for annotating the model
    - Agent monitors for inspecting and controlling agents

- Export and import functions (export data, save and restore state of model, make a movie)
- BehaviorSpace tool used to collect data from multiple runs of a model

Eclipse

Eclipse was developed in the 1990's by a subsidiary of IBM, called Object Technology International (OIT), to provide a Java-based replacement for the Smalltalk language created in the 1970's as part of the VisualAge family of integrated development products. It was released in January 2004 as an open-source Project by the Eclipse Foundation [454]. Eclipse is a software development platform composed of an integrated development environment (IDE) and an extensible plug-in system. By default, Eclipse is directed towards the development of applications in Java [455]. Nevertheless, the capabilities of Eclipse can be extended to handle other programming languages such as C, C++, COBOL, Python, Perl, PHP, Scala, and Ruby. Eclipse is also highly flexible, allowing users to write their own plug-in modules. As such, Eclipse can be perceived as everything from a Java IDE to a full-fledged stand-alone software development framework. More realistically, Eclipse is an open environment where any kind of software tools can plug in, thus providing the building blocks and infrastructure to create any user defined environment. It is thus widely used as a platform for doing native development, web design, service-oriented-architecture design, and embedded-software design. Eclipse offers a wide variety of examples of plug-in tools and environments that can be used as building blocks and integration points.

A list of the main features of the Eclipse software development environment is provided below:

- Multi-language programming functionalities on top of (and including) the runtime system
- Runtime system based on Equinox (OSGi standard compliant implementation [456])
- Use of plug-ins (functionality is not hard coded)

- Wide variety of plug-ins: UML plug-in for Sequence and other UML applications, DB Explorer plug-in
- Compatibility with typesetting languages like LaTeX
- Networking applications such as telnet, and database management systems
- Built-in incremental Java compiler and full model of the Java source files
- Advanced refactoring techniques and code analysis
- Use of a workspace, allowing external file modifications
- Possible implementation of widgets through the widget toolkit for Java SWT
- Possible translations into over a dozen natural languages [457]

MATLAB

MATLAB is a high-level language and interactive environment created in the late 1970's by Cleve Moler at the University of New Mexico [458]. Originally, Cleve Moler designed MATLAB for his students, in order to give them access to software libraries for numerical linear algebra, such as LINPACK and EISPACK, without having them learn to program in FORTRAN. But MATLAB soon spread to other universities and became popular in the applied mathematics community. As a consequence of this success, and with the help of Jack Little from the Stanford University and Steve Bangert, Cleve Moler founded MathWorks in 1984 to continue the development of MATLAB. MATLAB is now widely used in several domains of studies and in education, particularly for teaching linear algebra, numerical analysis and image processing.

In MATLAB, it is not necessary to declare variables, specify data types, and allocate memory. As such, MATLAB enables users to perform computationally intensive tasks faster than traditional programming languages such as C, C++, and Fortran. That is why it is widely used in applications such as signal and image processing, communications, control design, test and measurement, financial modeling and analysis, aerospace engineering, and computational biology.

A list of the key features of MATLAB, taken directly from the MathWorks website [459], is provided below:

- High-level language for technical computing
- Interactive development environment for managing code, files, and data
- Interactive tools for iterative exploration, design, and problem solving
- Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, and numerical integration
- Arithmetic operators, data structures, data types, debugging features, and object-oriented programming
- Support for entire data analysis process, from acquiring data, through preprocessing, visualization, and numerical analysis, to producing quality output
- Graphics features required to visualize engineering and scientific data, such as 2D and 3D plotting functions, 3D volume visualization functions, tools for interactively creating plots
- Ability to export results to all popular graphics formats
- Mathematical, statistical, and engineering functions to support all common engineering and science operations
- Functions for documenting and sharing work among users
- Functions for integrating MATLAB based algorithms with external applications and languages (C, C++, Fortran, Java, COM, and Microsoft Excel)
- Functions for distributing MATLAB algorithms and applications among several computer platforms
- Functions for deploying custom MATLAB algorithms and applications as stand-alone programs or software modules

Appendix I – Latitudes and Longitudes of the 256 Most Promising Positions

Position Index	Latitude (° North)	Longitude (° West)	X	Y	Position Index	Latitude (° North)	Longitude (° West)	X	Y
1	32.474	114.800	0	201	129	31.737	112.442	223	119
2	32.465	114.789	1	200	130	31.755	112.431	224	121
3	32.456	114.768	3	199	131	31.755	112.420	225	121
4	32.393	114.757	4	192	132	31.746	112.399	227	120
5	32.483	114.757	4	202	133	31.845	112.376	229	131
6	32.411	114.747	5	194	134	31.782	112.367	230	124
7	32.465	114.747	5	200	135	31.791	112.356	231	125
8	32.411	114.736	6	194	136	31.809	112.345	232	127
9	32.474	114.736	6	201	137	31.710	112.337	233	116
10	32.375	114.683	11	190	138	31.710	112.326	234	116
11	32.384	114.683	11	191	139	31.764	112.304	236	122
12	32.375	114.672	12	190	140	31.764	112.283	238	122
13	32.357	114.640	15	188	141	31.683	112.253	241	113
14	32.357	114.630	16	188	142	31.710	112.242	242	116
15	32.393	114.619	17	192	143	31.737	112.230	243	119
16	32.348	114.608	18	187	144	31.764	112.187	247	122
17	32.402	114.608	18	193	145	31.737	112.178	248	119
18	32.420	114.608	18	195	146	31.692	112.168	249	114
19	32.384	114.587	20	191	147	31.674	112.148	251	112
20	32.429	114.576	21	196	148	31.656	112.116	254	110
21	32.402	114.544	24	193	149	31.647	112.106	255	111
22	32.366	114.491	29	189	150	31.620	112.065	259	106
23	32.357	114.481	30	188	151	31.620	112.054	259	106
24	32.312	114.449	33	183	152	31.620	112.033	262	106
25	32.294	114.417	36	181	153	31.611	112.002	265	105
26	32.366	114.385	39	189	154	31.719	111.967	268	117
27	32.366	114.374	40	189	155	31.737	111.966	268	117
28	32.312	114.364	41	183	156	31.629	111.948	268	119
29	32.276	114.343	43	179	157	31.629	111.906	270	107
30	32.276	114.300	47	179	158	31.575	111.897	275	101
31	32.285	114.289	48	180	159	31.620	111.875	275	101
32	32.366	114.268	50	189	160	31.593	111.865	278	103
33	32.357	114.257	51	188	161	31.629	111.853	279	107
34	32.249	114.236	53	176	162	31.593	111.844	280	103
35	32.294	114.215	55	181	163	31.629	111.843	280	103
36	32.294	114.204	56	181	164	31.575	111.823	280	103
37	32.231	114.183	58	174	165	31.557	111.782	286	99
38	32.240	114.173	59	175	166	31.539	111.740	282	101
39	32.294	114.140	62	181	167	31.584	111.728	291	102
40	32.312	114.140	62	183	168	31.566	111.707	293	100

Position Index	Latitude (° North)	Longitude (° West)	X	Y	Position Index	Latitude (° North)	Longitude (° West)	X	Y
41	32.303	114.130	63	182	169	31.539	111.687	291	102
42	32.303	114.119	64	182	170	31.539	111.677	296	97
43	32.276	114.077	68	179	171	31.557	111.644	295	97
44	32.231	114.066	69	174	172	31.584	111.622	301	102
45	32.231	114.056	70	174	173	31.539	111.592	299	99
46	32.205	114.045	71	171	174	31.557	111.581	301	102
47	32.205	114.035	72	171	175	31.485	111.552	308	91
48	32.196	114.024	73	170	176	31.494	111.541	309	92
49	32.240	114.003	75	175	177	31.494	111.509	308	91
50	32.169	113.971	78	167	178	31.458	111.458	317	88
51	32.169	113.961	79	167	179	31.449	111.437	319	87
52	32.169	113.950	80	167	180	31.530	111.403	317	88
53	32.160	113.940	81	166	181	31.494	111.393	323	92
54	32.205	113.907	84	171	182	31.440	111.364	326	86
55	32.205	113.875	87	171	183	31.521	111.350	327	95
56	32.196	113.865	88	170	184	31.440	111.343	326	86
57	32.142	113.844	90	164	185	31.440	111.300	327	95
58	32.169	113.833	91	167	186	31.503	111.266	335	93
59	32.133	113.791	95	163	187	31.494	111.256	336	92
60	32.133	113.770	97	163	188	31.395	111.249	335	93
61	32.115	113.749	99	161	189	31.404	111.218	336	92
62	32.196	113.748	99	170	190	31.476	111.194	342	90
63	32.097	113.728	101	159	191	31.359	111.156	340	82
64	32.124	113.706	103	162	192	31.422	111.133	348	84
65	32.088	113.696	104	158	193	31.467	111.120	349	89
66	32.169	113.663	107	167	194	31.467	111.120	349	89
67	32.169	113.653	108	167	195	31.386	111.092	349	89
68	32.079	113.643	109	157	196	31.341	111.051	356	75
69	32.142	113.610	112	164	197	31.359	111.030	358	77
70	32.061	113.590	114	155	198	31.368	111.019	358	77
71	32.070	113.580	115	156	199	31.368	111.008	360	78
72	32.061	113.569	116	155	200	31.368	110.977	360	78
73	32.115	113.558	117	161	201	31.386	110.965	364	80
74	32.070	113.516	121	156	202	31.431	110.942	364	80
75	32.088	113.473	125	158	203	31.395	110.912	366	85
76	32.133	113.462	126	163	204	31.386	110.902	370	80
77	32.133	113.451	127	163	205	31.341	110.894	371	75
78	32.025	113.432	129	151	206	31.359	110.882	371	75
79	32.106	113.420	130	160	207	31.422	110.848	372	77
80	32.016	113.389	133	150	208	31.341	110.841	375	84
81	32.007	113.379	134	149	209	31.431	110.795	380	85
82	32.052	113.346	137	154	210	31.431	110.784	380	85

Position Index	Latitude (° North)	Longitude (° West)	X	Y	Position Index	Latitude (° North)	Longitude (° West)	X	Y
83	32.007	113.326	139	149	211	31.377	110.745	385	79
84	32.034	113.283	143	152	212	31.404	110.722	387	82
85	32.016	113.262	145	150	213	31.422	110.700	387	82
86	32.052	113.240	147	154	214	31.440	110.679	391	86
87	32.007	113.220	149	149	215	31.368	110.671	391	86
88	32.016	113.198	151	150	216	31.368	110.661	392	78
89	32.025	113.177	153	151	217	31.377	110.639	395	79
90	31.998	113.167	154	148	218	31.386	110.607	398	80
91	31.998	113.156	155	148	219	31.359	110.598	399	77
92	31.998	113.135	157	148	220	31.341	110.557	399	77
93	31.998	113.114	159	148	221	31.377	110.513	407	79
94	31.989	113.093	161	147	222	32.142	113.462	126	164
95	31.989	113.082	162	147	223	31.710	112.242	242	116
96	31.917	113.041	166	139	224	31.404	110.733	242	116
97	31.962	113.030	167	144	225	31.377	110.629	396	79
98	31.971	113.019	168	145	226	32.330	114.278	396	79
99	31.935	113.009	169	141	227	31.494	111.256	49	185
100	31.998	112.987	171	148	228	32.339	114.310	46	186
101	31.998	112.976	172	148	229	31.890	112.735	46	186
102	31.917	112.956	174	139	230	31.872	112.693	195	136
103	31.980	112.934	176	146	231	31.836	112.312	235	130
104	31.908	112.925	177	138	232	32.097	113.399	235	130
105	31.917	112.893	180	139	233	31.935	112.914	132	159
106	31.962	112.860	183	144	234	31.710	112.221	178	141
107	31.908	112.840	185	138	235	31.593	111.643	244	116
108	31.881	112.830	186	135	236	31.440	111.353	327	86
109	31.854	112.788	190	132	237	31.737	112.463	327	86
110	31.836	112.768	192	130	238	31.944	112.808	188	142
111	31.872	112.735	195	134	239	31.431	111.143	188	142
112	31.863	112.714	197	133	240	31.368	110.629	347	85
113	31.854	112.682	200	132	241	31.341	110.894	371	75
114	31.845	112.651	203	131	242	32.375	114.193	371	75
115	31.845	112.640	204	131	243	31.944	112.882	181	142
116	31.791	112.610	207	125	244	31.926	112.638	181	142
117	31.800	112.610	207	126	245	31.557	111.697	294	99
118	31.800	112.599	208	126	246	31.935	112.797	189	141
119	31.818	112.577	210	128	247	32.393	114.278	189	141
120	31.773	112.568	211	123	248	31.980	112.839	185	146
121	31.773	112.557	212	123	249	31.935	112.903	179	141
122	31.755	112.526	215	121	250	31.620	111.906	179	141
123	31.809	112.514	216	127	251	32.375	114.374	274	106
124	31.782	112.504	217	124	252	32.142	113.451	40	190

Position Index	Latitude (° North)	Longitude (° West)	X	Y	Position Index	Latitude (° North)	Longitude (° West)	X	Y
125	31.809	112.472	220	127	253	31.899	112.703	198	137
126	31.737	112.463	221	119	254	32.187	113.535	198	137
127	31.755	112.452	222	121	255	31.845	112.376	119	169
128	31.764	112.452	222	122	256	32.339	114.332	44	186

Appendix J – Space Filling DoE for the Modified GA (Left) and for the Modified PSO (Right)

Modified Genetic Algorithm				Modified Particle Swarm Optimization Algorithm			
S_Max	***MutRate***	***StopIte***	***PopSize***	***S_Max***	C_2	***IteNb***	***PopSize***
10	17.27	41	133	10	2.4	1705	71
12	36.82	71	83	12	2.1	1648	85
14	14.55	87	82	14	2.1	1770	182
16	25	100	142	16	2.4	1717	129
18	23.18	58	95	18	2.1	1952	176
20	28.64	63	145	20	2.2	1838	156
22	40	86	186	22	2.3	1790	108
23	44.55	42	177	23	2.4	1891	53
25	40.45	89	124	25	2.3	1927	92
27	16.36	72	179	27	2.3	1956	145
29	25.45	82	112	29	2.1	1786	70
31	26.36	36	64	31	2.1	1968	121
33	45.45	66	144	33	2.2	1640	126
35	29.09	41	168	35	2.4	1774	174
37	47.73	83	58	37	2.4	1851	130
39	9.09	47	180	39	2.3	1636	188
41	31.82	94	74	41	2.1	1624	173
43	40.91	43	127	43	2	1879	150
45	7.27	48	103	45	2.4	1915	179
46	11.82	92	130	46	2.2	1822	61
48	48.64	59	88	48	2.5	1818	82
50	13.18	55	55	50	2.5	1657	98
52	21.36	77	52	52	2.1	1737	139
54	37.73	60	192	54	2.1	1947	67
56	30.45	84	156	56	2.2	1689	58
58	13.64	70	105	58	2	1616	124
60	35	55	53	60	2.2	1749	189
62	45	33	71	62	2	1842	100
64	18.64	98	189	64	2.3	1600	74
66	33.64	61	106	66	2.4	1632	142
68	5.91	84	70	68	2.2	1855	115
69	27.27	38	109	69	2.5	1972	132
71	42.73	99	164	71	2.4	1976	77
73	5	70	152	73	2.3	1725	103
75	49.55	81	108	75	2.1	1863	191

77	23.64	53	197	77	2.4	1733	52
79	17.73	53	138	79	2.3	1919	180
81	33.18	99	115	81	2.3	1939	62
83	20.45	91	91	83	2.1	1988	167
85	36.36	82	200	85	2.3	1810	148
87	20	34	167	87	2	1669	76
89	47.27	46	165	89	2.4	1826	198
91	6.82	35	136	91	2.5	1612	194
93	43.64	76	155	93	2.5	1778	123
94	43.18	96	67	94	2.1	1794	56
96	30.91	75	73	96	2.1	1943	118
98	16.82	72	183	98	2.2	2000	133
100	28.18	67	148	100	2.2	1677	141
102	8.64	38	79	102	2.1	1681	168
104	19.09	87	139	104	2.2	1620	86
106	46.82	58	118	106	2.3	1653	192
108	22.27	51	56	108	2.4	1729	164
110	39.55	32	126	110	2.4	1871	159
112	7.73	65	76	112	2.4	1846	80
114	35.45	48	86	114	2	1814	136
116	21.82	63	100	116	2.2	1996	83
117	34.09	37	174	117	2.1	1741	106
119	37.27	80	114	119	2	1887	73
121	39.09	60	188	121	2.4	1665	89
123	32.27	93	158	123	2.3	1745	68
125	9.55	52	171	125	2.2	1867	94
127	11.36	31	198	127	2.4	1984	177
129	6.36	90	89	129	2.2	1907	152
131	49.09	75	77	131	2.3	1980	91
133	24.09	31	98	133	2.3	1895	135
135	8.18	92	176	135	2.3	1766	120
137	30	49	132	137	2.5	1923	117
139	19.55	79	62	139	2.2	1762	158
141	10.91	50	120	141	2.2	1903	200
142	10	73	135	142	2.3	1899	50
144	24.55	54	182	144	2.1	1697	55
146	48.18	95	147	146	2.3	1806	183
148	27.73	94	94	148	2.2	1604	147
150	18.18	36	153	150	2.1	1608	97
152	50	69	162	152	2.1	1673	197
154	26.82	79	194	154	2.3	1628	114
156	38.64	88	50	156	2.5	1685	138

158	32.73	68	85	158	2.4	1758	59
160	34.55	74	150	160	2.5	1721	195
162	35.91	30	65	162	2.1	1834	185
164	46.36	44	173	164	2	1754	88
165	15.91	39	61	165	2.3	1661	170
167	14.09	97	129	167	2	1701	153
169	41.36	85	191	169	2.5	1883	186
171	45.91	51	68	171	2.5	1802	109
173	41.82	40	121	173	2	1960	102
175	5.45	57	80	175	2.2	1992	171
177	22.73	67	123	177	2.1	1875	64
179	42.27	89	102	179	2	1964	161
181	44.09	65	117	181	2.1	1830	127
183	12.27	78	92	183	2.2	1709	112
185	20.91	62	59	185	2.2	1931	111
187	25.91	46	97	187	2.4	1911	79
188	15	77	170	188	2.3	1693	65
190	12.73	34	111	190	2.4	1644	95
192	31.36	43	161	192	2.2	1859	162
194	15.45	45	195	194	2.4	1782	155
196	29.55	96	159	196	2.4	1935	144
198	10.45	56	141	198	2.3	1798	105
200	38.18	64	185	200	2.2	1713	165

Appendix K – Custom DoE for the Modified GA (Left) and for the Modified PSO (Right)

Modified Genetic Algorithm				Modified Particle Swarm Optimization Algorithm			
S_Max	*MutRate*	*StopIte*	*PopSize*	*S_Max*	C_2	*IteNb*	*PopSize*
10	5	90	50	10	2.5	2000	110
10	10	30	200	10	2.4	2000	200
10	15	90	80	10	2	1700	200
10	20	50	200	10	2.1	1800	170
10	25	110	50	10	2.2	1900	50
10	30	50	80	10	2.4	1900	80
10	35	70	170	10	2.3	1700	80
10	40	110	110	10	2	1800	170
10	45	70	170	10	2.2	1600	140
10	50	30	140	10	2.3	1600	50
30	5	110	80	30	2.3	1700	200
30	10	70	110	30	2	2000	200
30	15	90	200	30	2.1	2000	50
30	20	30	50	30	2.3	1600	170
30	25	70	200	30	2.2	1800	80
30	30	50	170	30	2.4	1700	110
30	35	30	170	30	2.5	1800	80
30	40	50	140	30	2.2	1600	110
30	45	90	140	30	2	1900	170
30	50	110	80	30	2.4	1900	140
50	5	30	110	50	2.2	1900	80
50	10	90	170	50	2.4	2000	170
50	15	50	140	50	2.5	1900	140
50	20	110	50	50	2.3	1800	110
50	25	50	170	50	2.3	1700	200
50	30	30	140	50	2.5	1800	50
50	35	70	50	50	2.1	1600	80
50	40	70	80	50	2.1	1700	140
50	45	110	200	50	2.4	2000	170
50	50	90	110	50	2	1600	110
70	5	110	140	70	2	1900	110
70	10	90	80	70	2.1	1900	200
70	15	70	110	70	2.3	1800	200
70	20	50	140	70	2.5	1700	170

70	25	30	50	70	2.5	2000	50
70	30	90	170	70	2.1	1800	140
70	35	110	80	70	2.4	1600	110
70	40	30	200	70	2.2	2000	140
70	45	70	50	70	2	1700	80
70	50	50	170	70	2.4	1600	50
90	5	90	140	90	2.5	1700	140
90	10	30	80	90	2.1	2000	170
90	15	110	170	90	2.2	1700	50
90	20	70	110	90	2.2	1900	200
90	25	110	140	90	2.5	1900	110
90	30	70	110	90	2.4	1800	200
90	35	50	50	90	2	1600	50
90	40	70	80	90	2	1800	80
90	45	90	50	90	2.3	2000	80
90	50	30	200	90	2.1	1600	140
110	5	50	170	110	2.1	1700	110
110	10	110	110	110	2.4	1900	200
110	15	30	80	110	2	1900	170
110	20	90	200	110	2.2	1600	80
110	25	70	140	110	2.1	1600	50
110	30	30	110	110	2.5	1800	110
110	35	90	140	110	2.2	2000	200
110	40	110	200	110	2.3	1700	140
110	45	50	80	110	2.3	1800	170
110	50	70	50	110	2.5	2000	50
130	5	30	200	130	2.3	1800	140
130	10	50	50	130	2	2000	140
130	15	110	170	130	2.1	1900	50
130	20	70	140	130	2	1600	200
130	25	30	80	130	2.4	1700	80
130	30	70	80	130	2.1	1900	110
130	35	90	110	130	2.3	1600	170
130	40	90	50	130	2.2	1800	200
130	45	50	200	130	2.4	2000	80
130	50	110	110	130	2.5	1700	170
150	5	70	80	150	2.5	1600	200
150	10	70	200	150	2.3	1900	80
150	15	50	110	150	2.1	2000	200
150	20	30	170	150	2.4	1600	140
150	25	90	110	150	2	1700	50
150	30	110	140	150	2.4	1800	110

150	35	110	200	150	2.1	1900	170
150	40	50	50	150	2.2	1700	170
150	45	30	140	150	2.3	2000	110
150	50	90	170	150	2	1800	140
170	5	70	170	170	2	2000	140
170	10	70	140	170	2.2	1700	110
170	15	30	50	170	2.2	1800	50
170	20	90	80	170	2.5	1600	200
170	25	50	110	170	2.5	1600	80
170	30	110	50	170	2.3	1900	50
170	35	30	140	170	2.4	1700	170
170	40	90	110	170	2.1	1800	110
170	45	110	170	170	2.3	1900	140
170	50	50	200	170	2	2000	80
190	5	50	110	190	2.4	1800	50
190	10	110	140	190	2.1	1700	80
190	15	70	200	190	2.5	1600	140
190	20	110	80	190	2.2	2000	110
190	25	90	170	190	2.2	1600	170
190	30	90	200	190	2.1	1800	200
190	35	50	80	190	2.5	1900	80
190	40	30	170	190	2.3	2000	110
190	45	30	110	190	2	1700	50
190	50	70	50	190	2.4	1900	140

Appendix L – Relative Coverage Effectiveness of Each Type of Sensor System at Each Promising Position (Expressed in %)

Position Index	HCR	MCR	LCR	HCC	MCC	LCC	Position Index	HCR	MCR	LCR	HCC	MCC	LCC
1	1.43	1.73	2.37	0.7	2.05	2.18	129	1.47	1.63	1.66	0.42	1.31	1.78
2	1.36	1.62	2.16	0.71	1.92	2	130	3.1	2.99	1.73	0.44	1.24	2.33
3	1.22	1.43	1.8	0.71	1.63	1.71	131	3.05	3.23	2.23	0.42	1.57	2.93
4	1.24	1.44	1.78	0.7	1.59	1.71	132	2.8	3.05	2.48	0.63	1.9	2.86
5	2.24	2.55	2.48	0.56	1.89	2.84	133	1.2	1.2	1.17	0.52	0.94	1.29
6	1.16	1.34	1.62	0.69	1.47	1.56	134	0.66	0.72	0.83	0.41	0.76	0.79
7	2.08	2.39	2.61	0.67	2	2.76	135	0.88	0.98	1.08	0.44	0.96	1.06
8	1.12	1.28	1.53	0.68	1.35	1.49	136	0.88	0.94	1.19	0.47	1.05	1.11
9	2.4	2.68	2.67	0.63	2	2.95	137	0.84	0.97	1.22	0.44	1.05	1.2
10	2.58	2.85	2.56	0.7	1.98	2.89	138	1.1	1.31	1.43	0.45	1.18	1.48
11	2.42	2.69	2.57	0.67	2.03	2.8	139	1.32	1.57	1.59	0.44	1.28	1.69
12	2.22	2.48	2.53	0.7	2.03	2.68	140	1.32	1.47	1.46	0.46	1.24	1.59
13	1.44	1.62	1.94	0.64	1.6	1.86	141	2.9	3.11	2.5	0.45	1.85	3.01
14	1.42	1.6	1.92	0.62	1.55	1.85	142	3.65	3.87	2.87	0.5	1.95	3.66
15	2.24	2.49	2.2	0.44	1.65	2.64	143	4.25	4.47	3	0.52	2.03	4.12
16	1.31	1.48	1.85	0.53	1.57	1.78	144	4.28	4.43	2.95	0.49	2.02	3.96
17	2.92	3.25	2.33	0.37	1.78	2.84	145	4.3	4.39	2.85	0.46	2.02	3.85
18	3.28	3.64	2.27	0.52	1.8	2.96	146	3.15	3.08	2.67	0.58	1.96	3.06
19	2.3	2.58	2.1	0.37	1.48	2.67	147	2.79	2.77	2.2	0.57	1.67	2.66
20	3.1	3.23	1.83	0.52	1.44	2.3	148	1.49	1.65	1.46	0.44	1.16	1.7
21	2.84	2.83	1.82	0.54	1.38	2.35	149	3.66	3.63	2.52	0.6	1.95	3.11
22	2.88	2.99	2.34	0.58	1.85	2.7	150	5.91	5.42	3.32	0.47	2.2	4.46
23	2.82	2.87	2.21	0.56	1.7	2.62	151	2.64	2.67	1.43	0.46	1.1	1.79
24	2.79	2.8	2.19	0.5	1.66	0.26	152	5.11	4.82	2.93	0.44	2.01	4
25	2.79	2.79	2.17	0.49	1.61	2.58	153	4.91	4.86	2.91	0.34	1.81	4.08
26	2.49	2.47	1.95	0.57	1.48	2.27	154	4.71	4.51	2.81	0.46	1.87	3.76
27	1.16	1.31	1.54	0.46	1.37	1.44	155	4.25	4.38	2.95	0.45	1.93	3.86
28	1.23	1.4	1.63	0.42	1.42	1.55	156	3.85	4.28	3.04	0.44	2.12	4
29	1.02	1.12	1.25	0.43	1	1.31	157	2.94	3.45	2.8	0.39	1.98	3.58
30	1.41	1.57	1.27	0.42	0.93	1.47	158	2.43	2.7	2.73	0.47	1.91	3.04
31	1.61	1.7	1.49	0.42	1.14	1.69	159	0.77	0.84	1	0.5	0.93	0.98
32	1.76	1.89	1.61	0.43	1.19	1.87	160	3.94	3.62	2.11	0.4	1.51	2.73
33	5.01	4.4	2.3	0.49	1.66	3.18	161	0.81	0.95	1.08	0.54	0.94	1.06
34	4.58	4.17	2.11	0.41	1.45	2.95	162	0.91	1.1	1.27	0.54	1.13	1.24
35	1.58	1.75	1.8	0.44	1.37	1.95	163	1.23	1.45	1.51	0.53	1.25	1.55
36	2.6	2.7	1.94	0.55	1.45	2.48	164	1.5	1.79	1.91	0.48	1.54	2.13

Position Index	HCR	MCR	LCR	HCC	MCC	LCC	Position Index	HCR	MCR	LCR	HCC	MCC	LCC
37	0.26	2.78	0.2	0.54	1.46	2.48	165	1.61	1.88	1.97	0.49	1.51	2.17
38	2.65	2.78	1.95	0.52	1.47	2.49	166	1.21	1.37	1.53	0.51	1.2	1.58
39	1.69	1.89	2.01	0.42	1.54	2.11	167	1.64	1.84	1.76	0.56	1.38	2.06
40	1.75	1.95	2.01	0.43	1.47	2.24	168	2.56	2.7	1.77	0.48	1.33	2.25
41	1.84	2.08	2.01	0.44	1.45	2.25	169	2.48	2.63	1.93	0.46	1.47	2.31
42	4.2	3.95	2.1	0.47	1.52	2.81	170	2.23	2.39	1.7	0.44	1.32	2.02
43	3	3.2	2.02	0.48	1.48	2.44	171	3.51	3.25	1.96	0.55	1.48	2.5
44	3.65	3.75	2.25	0.49	1.59	2.75	172	4.27	4.22	2.76	0.62	2.05	3.47
45	4.15	4.25	2.37	0.52	1.63	3.25	173	4.04	3.98	2.66	0.43	1.85	3.54
46	4.98	4.66	2.6	0.53	1.84	3.48	174	5.87	5.65	3.88	0.61	2.67	5
47	4.82	4.55	2.52	0.56	1.81	3.37	175	2.89	3.06	2.16	0.47	1.59	2.76
48	3.59	3.72	2.32	0.58	1.75	3.01	176	2.89	3.09	2.37	0.47	1.75	2.95
49	3.73	4.14	2.58	0.55	1.85	3.42	177	2.89	3.15	2.55	0.46	1.83	3.09
50	4.4	4.41	2.8	0.52	2	3.66	178	2.26	2.59	2	0.44	1.43	2.63
51	4.86	4.79	3.09	0.48	2.18	4.08	179	2.4	2.32	1.66	0.42	1.15	2.13
52	4.9	0.48	0.31	0.47	0.22	4.25	180	3.39	3.6	2.29	0.48	1.54	3.21
53	4.98	4.96	3.24	0.46	2.16	4.35	181	2.66	3.13	2.62	0.46	1.87	3.33
54	4.71	4.73	3.2	0.51	2.15	4.19	182	1.57	1.74	1.48	0.44	1.13	1.78
55	3.95	4.02	2.76	0.55	1.89	3.7	183	3.23	3.52	1.98	0.3	1.35	2.84
56	0.83	0.95	1.25	0.48	1.15	1.14	184	1.62	1.81	1.47	0.4	1.15	1.7
57	2.59	2.88	2.6	0.58	1.91	3.12	185	1.31	1.38	1.25	0.36	1.01	1.33
58	2.75	3.09	2.75	0.56	1.95	3.39	186	2.1	1.99	1.62	0.36	1.21	1.84
59	3.1	3.25	2.85	0.56	2.09	3.52	187	1.52	1.23	1.32	0.41	1	1.35
60	3.27	3.56	2.96	0.54	2.17	3.61	188	3.59	3.28	2.24	0.47	1.65	2.89
61	2.75	2.95	2.65	0.53	2.05	3.1	189	4.12	3.99	2.3	0.46	1.54	3.1
62	2.42	2.73	2.48	0.51	1.91	2.83	190	5.01	4.9	2.92	0.58	2.13	3.92
63	1.83	2.04	1.98	0.49	1.54	2.13	191	4.86	4.58	2.76	0.54	1.97	3.55
64	1.14	1.32	1.28	0.44	1.02	1.31	192	4.28	4.04	2.14	0.47	1.64	2.83
65	2	2.28	2.03	0.55	1.61	2.32	193	4.36	4.28	2.47	0.47	1.75	3.48
66	2.38	2.59	2.34	0.59	1.84	2.44	194	4.47	4.65	2.73	0.46	1.83	3.84
67	2.43	2.73	2.46	0.63	1.95	2.7	195	2.68	3.13	2.59	0.45	1.86	3.23
68	1.6	1.81	1.84	0.57	1.51	1.95	196	0.71	0.82	0.96	0.44	0.81	0.95
69	2.08	2.28	2.13	0.51	1.61	2.38	197	0.78	0.91	0.97	0.41	0.74	1.08
70	1.95	2.22	2.52	0.66	1.97	2.71	198	0.76	0.9	1.08	0.38	0.97	1.05
71	1.48	1.72	1.83	0.49	1.41	1.97	199	1.21	1.45	1.73	0.35	1.39	1.79
72	2.01	1.96	1.58	0.52	1.37	1.61	200	1.48	1.75	1.56	0.37	1.15	1.91
73	3.19	2.58	1.94	0.5	1.56	2.65	201	1.68	1.97	1.55	0.4	1.18	1.98
74	3.31	3.31	2.59	0.49	1.94	3.07	202	2.69	2.89	1.69	0.4	1.18	2.35
75	2.99	3.09	2.25	0.41	1.51	2.82	203	3.57	3.47	1.95	0.43	1.38	2.59

Position Index	HCR	MCR	LCR	HCC	MCC	LCC	Position Index	HCR	MCR	LCR	HCC	MCC	LCC
76	3.22	2.95	2.12	0.49	1.64	2.53	204	4.29	4.08	2.14	0.46	1.54	2.89
77	3.53	3.43	2.19	0.49	1.63	2.74	205	4.5	4.25	2.25	0.48	1.64	3.15
78	3.84	3.65	2.25	0.5	1.61	2.95	206	3.96	3.87	2.59	0.5	1.76	3.19
79	2.77	2.63	2.15	0.56	1.66	2.46	207	3.38	3.33	2.73	0.51	1.91	3.2
80	2.79	2.66	2.22	0.64	1.79	2.6	208	4.35	4.52	3.32	0.53	2.24	4.3
81	3.15	2.95	2.31	0.6	1.69	0.29	209	2.38	2.75	2.72	0.62	2.11	3.11
82	3.43	3.54	2.32	0.47	1.62	3.11	210	2.04	2.18	2.08	0.54	1.59	2.32
83	1.66	1.66	1.54	0.49	1.24	1.74	211	2.43	2.58	2.17	0.54	1.66	2.6
84	2.35	2.1	1.85	0.52	1.57	2.42	212	3.02	3.01	2.12	0.62	1.58	2.64
85	2.58	2.85	2.38	0.54	1.75	2.97	213	3.78	3.86	2.65	0.59	1.97	3.33
86	2.38	2.62	2.46	0.62	1.81	2.81	214	3.19	3.24	2.25	0.53	1.77	3.06
87	1.77	1.87	2.1	0.52	1.56	2.07	215	2.78	2.69	1.97	0.48	1.52	2.41
88	1.26	1.27	1.21	0.46	1.07	1.26	216	1.94	1.83	1.33	0.52	1.08	1.41
89	1.99	2.3	2.29	0.6	1.88	2.45	217	1.56	1.75	1.43	0.53	1.04	1.8
90	1.32	1.57	1.63	0.54	1.38	1.78	218	1.95	2.4	1.87	0.56	1.59	2.37
91	2.57	2.66	1.95	0.48	1.49	2.65	219	2.16	2.5	2.46	0.59	1.9	2.71
92	3.24	3.45	2.12	0.41	1.52	2.84	220	2.81	2.72	1.74	0.4	1.27	2.19
93	4.56	4.16	2.07	0.44	1.45	2.91	221	2.74	2.89	1.92	0.35	1.35	2.28
94	1.08	1.27	1.45	0.52	1.26	1.46	222	1	1.19	1.26	0.4	0.95	1.43
95	2.85	2.85	1.75	0.48	1.36	1.89	223	1.29	1.51	1.12	0.36	0.89	1.42
96	3.26	3.76	1.95	0.47	1.44	2.76	224	1.77	1.75	1.25	0.46	0.98	1.38
97	4.32	4.26	2.16	0.44	1.49	3.13	225	2.44	2.39	1.65	0.44	1.37	1.96
98	3.46	3.67	3.05	0.62	2.33	3.54	226	3.06	2.95	2.03	0.42	1.51	2.65
99	2.6	2.93	2.61	0.67	2.04	3.02	227	2.76	2.45	1.76	0.43	1.34	0.21
100	2.43	2.6	2.21	0.63	1.73	2.5	228	2.44	1.95	1.36	0.44	1.04	1.64
101	2.48	2.69	2.28	0.63	1.79	2.76	229	3.03	2.85	1.8	0.48	1.3	2.19
102	2.5	2.74	2.35	0.63	1.82	2.39	230	1.08	1.24	1.31	0.48	1.09	1.4
103	2.53	2.79	2.43	0.63	1.81	2.87	231	1.25	1.49	1.45	0.41	1.07	1.67
104	3.49	3.72	2.38	0.52	1.69	3.13	232	2.32	2.44	2.09	0.4	1.44	2.61
105	3.47	3.65	2.38	0.56	1.71	3.12	233	2.86	2.96	2.49	0.5	1.57	3.18
106	4.42	4.21	2.43	0.65	1.79	3.26	234	3.13	3.29	2.85	0.52	1.99	3.48
107	3.25	3.21	2	0.5	1.49	2.53	235	3.05	3.21	2.87	0.57	2.11	3.29
108	3.86	3.57	2	0.48	1.5	2.49	236	1.58	1.72	1.65	0.46	1.43	1.68
109	4.26	3.95	2.05	0.46	1.52	2.72	237	0.69	0.64	0.65	0.36	0.53	0.69
110	4.34	3.98	2.2	0.5	1.64	2.88	238	1.36	1.54	1.39	0.54	1.16	1.64
111	4.77	4.4	2.35	0.48	1.7	3.45	239	1.71	1.98	1.75	0.47	1.34	2.11
112	5.16	4.69	2.5	0.43	1.72	3.55	240	2.27	2.56	1.81	0.46	1.39	2.35
113	5.22	4.69	2.5	0.44	1.76	3.53	241	1.57	1.8	1.49	0.36	1.14	1.94
114	5.28	4.71	2.51	0.45	1.77	3.48	242	0.86	1.02	1.06	0.37	0.94	1.11

Position Index	HCR	MCR	LCR	HCC	MCC	LCC	Position Index	HCR	MCR	LCR	HCC	MCC	LCC
115	5.86	5.4	2.98	0.54	2.07	4.1	243	3.01	3.37	2.4	0.58	1.72	3.23
116	4.02	3.89	2.55	0.53	1.93	3.11	244	0.5	0.58	0.75	0.36	0.67	0.71
117	3.08	3.11	2.04	0.53	1.68	2.44	245	0.66	0.72	0.96	0.41	0.86	0.87
118	2.03	2.18	1.82	0.49	1.48	2.19	246	0.99	1.08	1.16	0.43	0.94	1.17
119	1.5	1.62	1.84	0.6	1.53	1.9	247	2.92	2.89	2.11	0.44	1.52	2.67
120	2.23	2.54	2.43	0.46	1.81	2.68	248	4.94	4.89	2.75	0.52	1.98	3.67
121	2.42	2.75	2.2	0.36	1.54	2.74	249	3.01	2.78	1.56	0.48	1.22	1.93
122	0.91	1.08	1.32	0.44	1.1	1.31	250	4.18	4.31	2.65	0.45	1.73	3.75
123	4.07	4.18	2.42	0.51	1.7	3.3	251	5.34	5.43	3.67	0.62	2.56	4.88
124	4.22	4.28	2.49	0.51	1.74	3.39	252	2.38	2.59	1.82	0.44	1.37	2.41
125	4.4	4.4	2.55	0.52	1.79	3.46	253	1.27	1.28	1.24	0.43	1.09	1.3
126	0.69	0.82	0.88	0.44	0.78	0.87	254	1.04	1.12	1.1	0.4	0.91	1.14
127	2.5	2.67	1.5	0.44	1.2	1.9	255	1	1.11	1.16	0.38	0.92	1.16
128	0.73	0.85	0.96	0.42	0.8	0.92	256	1.81	2.14	2.45	0.54	1.87	2.56

Appendix M – Latitudes and Longitudes of the Common Promising Locations at Which Detection Systems are Preferentially Located in the Single Type Systems Recursive Optimization Approach Across all System Types

Position Index	Latitude (° North)	Longitude (° West)
1	32.474	114.8
3	32.456	114.768
8	32.474	114.736
9	32.411	114.736
28	32.366	114.374
40	32.24	114.173
42	32.312	114.14
43	32.294	114.14
44	32.303	114.13
49	32.231	114.066
52	32.205	114.045
58	32.169	113.961
61	32.169	113.95
63	32.16	113.94
77	32.088	113.696
81	32.079	113.643
90	32.088	113.473
95	32.106	113.42
101	32.007	113.326
106	32.052	113.24
111	31.998	113.167
113	31.998	113.156
123	31.998	112.987
132	31.881	112.83
149	31.809	112.514
150	31.782	112.504
157	31.755	112.431
176	31.665	112.116
209	31.494	111.509

REFERENCES

[1] "Engineers' Council for Professional Development," *Science,* Vol. 94, Issue 2446, pp. 456

[2] Engineers' Council for Professional Development , "Canons of ethics for engineers," 1947

[3] Encyclopaedia Britannica: definition of *"Engineering"*

[4] Office of Homeland Security, "National Strategy for Homeland Security," July 2002

[5] Office of Homeland Security, "The National Strategy for Physical Protection of Critical Infrastructures and Key Assets," February 2003

[6] New Defence Agenda, "Strategic Priorities for protecting Europe's Infrastructure against terrorism," June 2005

[7] Commission of the European Communities, "On a European Program for Critical Infrastructure Protection," November 2005

[8] Congress USA Patriot Act, Section 1016 (e)

[9] Homeland Security Presidential Directive 7

[10] Ministère de la Défense, "La Défense contre le terrorisme, une priorité du ministère de la défense," October 2005

[11] Congressional Research Service, "Risk Management and CIP: Assessing, Integrating, and Managing Threats, Vulnerabilities and Consequences," *Report for Congress*, September 2004

[12] Office of Border Patrol, U.S. Customs and Border Protection, "National Border Patrol Strategy", September 2004

[13] Haddal, C. C., "Border Security: The Role of the U.S. Border Patrol", *Congressional Research Service Report for Congress*, August 2011

[14] U.S. Customs and Border Protection Fiscal Year 2009-2014 Strategic Plan, "Secure Borders, Safe Travel, Legal Trade", July 2009

[15] Rosenblum, M. R., "Border Security: Immigration Enforcement Between Ports of Entry", *Congressional Research Service Report for Congress*, January 2012

[16] U.S. Department of Transportation, Research and Special Programs Administration, "Spatial information technologies in Critical Infrastructure Protection, A Research Agenda In CIP," *National Consortium on Remote Sensing in Transportation (NCRS)*

[17] U.S. Customs and Border Protection, Department of Homeland Security, "Testimony of Michael J. Fisher, Chief, United States Border Patrol, Michael Kostelnik, Assistant Commissioner, Office of Air and Marine, Mark S. Borkowski, Assistant Commissioner, Office of Technology Innovation and Acquisition" *before the House Committee on Homeland Security Subcommittee on Border and Maritime Security*, March 2011, Washington DC

[18] European Commission, "Critical Infrastructures Protection as part of the fight against Terrorism," October 2004

[19] John L. Clarke, "Securing the Homeland," September 2003

[20] 1st article of the French 1959 ordinance

[21] Congressional Research Service, "European Approaches to Homeland Security and Counterterrorism," *Report for Congress*, July 2006

[22] European Commission Enterprise and Industry, "Security Research and Development"

[23] European Commission Enterprise and Industry, "PASR 2006: 15 new security research projects to combat terrorism", October 2006, Brussels

[24] New Defence Agenda, "Strategic Priorities for protecting Europe's Infrastructure against terrorism," June 2005

[25] Euresearch, Swiss guide to European research & innovation, "R&D Guide, FP7 Cooperation, Security", March 2012

[26] Center for the Study of Democracy, "Better Management of EU Borders Through Cooperation, Study to Identify Best Practices on the Cooperation Between Border Guards and Customs Administrations Working at the External Borders of the EU", 2011

[27] European Commission, "Prevention, Preparedness and Response to Terrorist Attacks", October 2004

[28] European Union Council, "EU Solidarity Programme on the consequences of terrorist threats and attacks", December 2004, Brussels

[29] Commission of the European Communities, "European Programme for Critical Infrastructure Protection", December 2006, Brussel

[30] Commission of the European Communities, “Critical Infrastructure Warning Information Network (CIWIN)”, October 2008, Brussels

[31] European Council Meeting in Laeken, “Presidency Conclusions“, December 2001

[32] Commission Communication, “Towards integrated management of the external borders of the Member States of the European Union“, May 2002, Brussels

[33] Commission Communication, “The Hague Programme: ten priorities for the next five years. The partnership for European renewal in the field of Freedom, Security and Justice“, May 2005, Brussels

[34] European Commission, Directorate General of Taxation and Customs Union, Customs Policy and Procedures, “Customs Code Committee, Import/Export Formalities Section – Guidelines on Export and Exit in the Context of Regulation (EC) No. 648/2005”, October 2012, Brussels

[35] “Treaty of Amsterdam amending the Treaty of the European Union, the Treaties establishing the European Communities and certain related acts, commonly known as the Amsterdam Treaty”, October 1997

[36] Vienna European Union Council, “Vienna Action Plan”, *Council and Commission Action Plan on how best to implement the provisions of the Treaty of Amsterdam on the creation of an area of freedom, security and justice*, December 1998, Vienna

[37] European Union Council, “Convention on mutual assistance and cooperation between customs administrations (Naples II)”, *Council Act 98/C 24/01, on the basis of Article K3 of the Treaty on European Union*, December 1997, Brussels

[38] European Union Council, “Convention on the use of information technology for customs purposes (CIS Convention)”, *Council Act 95/C316/02*, July 1995, Brussels

[39] European Union Council, “Strategy for the evolution of the Customs Union”, *Communication from the Commission to the Council, the European Parliament, the Economic and Social Committee and the Committee of the Regions*, April 2008, Brussels

[40] European Union Council, “European Agency for the management of External Borders – Frontex”, *Council Regulation on Establishing a European Agency for the Management of Operational Cooperation at the External Borders of the Member States of the European Union*, October 2004

[41] Clarke, D. J., Davis, E., and Varco, A. G., “Surveillance of Borders Coastlines and Harbours (SOBCAH) A European Commission Preparatory Action on Security Research”, Electro-Optical and Infrared Systems: Technology and Applications V,

edited by David A. Huckridge, Reinhard R. Ebert, *SPIE*, Vol. 7113, 2008. doi: 10.1117/12.801996

[42] Ackerman, G., Abhayaratne, P., Bale, J., Bhattacharjee, A., Blair, C., Hansell, L., Jayne, A., Kosal, M., Lucas, S., Moran, K., Seroki, L., and Vadlamudi, S., “Assessing Terrorist Motivations for Attacking Critical Infrastructure”, *University of California, Lawrence Livermore National Laboratory for the U.S. Department of Energy, Contract W-7405-Eng-48*, January 2007

[43] National Security Presidential Directive (NSPD) 47, “National Strategy for Aviation Security,” June 2006

[44] Critical Infrastructure Protection Information Center, United States Fire Administration, “The DODA Process, Job Aid,” May 2002

[45] Dr Ruth David, “Homeland Security Technologies: Creating an Asymmetric advantage,” April 2002

[46] Thales Group

[47] Rinaldi, S. M., “Modeling and Simulating Critical Infrastructures and Their Interdependencies,” *Sodia National Laboratories, 37th Hawaii International Conference on System Sciences*, 2004

[48] Thesaurus Dictionary: definition of *“Methodology”*

[49] Wolthusen, S. D., “GIS-based Command and Control Infrastructure for CIP,” *First IEEE International Workshop on Critical Infrastructure Protection (IWCIP’05)*, 2005

[50] DoD dictionary: definition of *“System”*

[51] International Council on Systems Engineering (INCOSE) Technical Board, Systems Engineering Handbook, International Council on Systems Engineering, INCOSE-TP-2003-002-03, Version 3.0, June 2006

[52] IEEE Std 1233, Guide for Developing System Requirements Specifications

[53] SMC Systems Engineering Primer & Handbook, Space & Missile Systems Center, U.S. Air Force, Second Edition, 15 January 2004

[54] Cotsaftis, M., “What makes a system complex? An approach to self organization and emergence,” *Emergent Properties in Natural and Artificial Complex Systems (EPNACS’2007), Dresden, Germany*, October 2007

[55] Laplace, P. S., “Oeuvres Complètes,” *Part 6, Gauthiers−Villard*, Paris, 1878-1912

[56] Lagrange, J. L., "Mécanique Analytique," *Volume I-II, Ed. Albert Blanchard, Paris*, 1788

[57] Aristotle, "Physics," Part I, Chap.9

[58] Anderson, P. W., "More is Different," *Science*, Vol. 177, pp.393-396, 1972

[59] Sheard, S. A., "Definition of the Sciences of Complex Systems," *INSIGHT, International Council on Systems Engineering, Seattle, Washington*, Vol. 9, No. 1, pp.25, October 2006

[60] Haskins, C., "Systems Engineering Handbook: A Guide for Life-Cycle Processes and Activities," *International Council on Systems Engineering*, 2006

[61] University of Michigan, "About the Science of Complexity"

[62] Maier, J. R. A., Fadel, G. M., "Understanding the Complexity of Design," in *"Complex Engineered Systems: Science Meets Technology,"* Braha, D., Minai, A. A., and Bar-Yam, Y., Cambridge, Massachusetts: Springer NECSI, 2006

[63] Norman, D. O., and Kuras, M. L., "Engineering Complex Systems," in *"Complex Engineered Systems: Science Meets Technology,"* Braha, D., Minai, A. A., and Bar-Yam, Y., Cambridge, Massachusetts: Springer NECSI, 2006

[64] Bar-Yam, Y., "Engineering complex systems: multiscale analysis and evolutionary engineering," in *"Complex Engineered Systems: Science Meets Technology,"* Braha, D., Minai, A. A., and Bar-Yam, Y., Cambridge, Massachusetts: Springer NECSI, 2006

[65] Crisp, H., and Ewald, B., "Capability Engineering for Systems of Systems: A Coalition Perspective," *INCOSE INSIGHT*, Vol. 8, No. 1, pp. 3, October 2005

[66] Clare, T. A., "The Engineering and Acquisition of Systems of Systems," in *3rd Annual Systems Engineering & Supportability Conference, Department of Defense, Research Development and Acquisition Office*, October 2000

[67] Maier, M. W., "Architecting Principles for Systems of Systems," *Sixth Annual International Symposium, International Council on Systems Engineering, Boston, MA*, 1996

[68] Hatley, D., Hruschka, P., and Pirbhai, I., "Process for System Architecture and Requirements Engineering," *Dorset House Publishing Company*, 2000

[69] Zachman, J. A., "A framework for information systems architecture," *IBM Systems Journal*, Vol. 26, No. 3, 1987

[70] NASA Systems Engineering Handbook, June 1995

[71] Systems Engineering Fundamentals, January 2001

[72] Sheard, S. A., "Principles of Complex Systems for Systems Engineering," *Third Millenium Systems LLC*, 2006

[73] Keating, C., Rogers, R., Unal, R., Dryer, D., Sousa-Poza, A., Safford, R., Peterson, W., and Rabadi, G., "System of Systems Engineering," *Old Dominion University, Engineering Management Journal, the circumstances within which the system-of-systems problem is embedded*, Vol. 15, No. 3, September 2003

[74] Martin, J. N., "Systems Engineering Guidebook: A Process for Developing Systems and Products," *CRC Press*, 1996

[75] Defense Systems Management College, "Systems Engineering Fundamentals," *Defense Acquisition University Press*, 1990

[76] Simon, H. A., "The New Science of management Decision," *Prentice Hall*, 1977

[77] Schön, D., "Toward a New Epistemology of Practice," chapter in *Strategic Perspectives on Planning*, Edition Checkoway , B. Lexington, pp. 231-250, 1986

[78] Mitroff, I., "Smart Thinking for Crazy Times," *Berrett-Koehler*, 1998

[79] Jackson, M. C., "Systems Methodology for the Management Sciences," *Perseus*, 1991

[80] Checkland, P., "Systems Thinking, Systems Practice," *Second Edition, Wiley*, 1999

[81] Alexander, R., Hall-May, M., and Kelly, T., "Characterisation of Systems of Systems Failures," *22nd International System Safety Conference, University of York*, 2004

[82] Despotou, G., Alexander, R., and Hall-May, M., "Key Concepts and Characteristics of Systems of Systems," *Technology Repository DARP-HIRTS Strand 2, University of York, Department of Computer Science*, May 2003

[83] Cantor, M., "Thoughts on Functional Decomposition," IBM online magazine *"The Rational Edge,"* 2003

[84] Pahl, G., and Beitz, W., "Engineering Design," *London: Springer-Verlag*, 1996

[85] Koller, R., "Konstruktionslehre fuer den Maschinenbau," *Berlin: Springer-Verlag*, 1985

[86] Ritchey, T., "General Morphological Analysis, a general method for non-quantified modeling," adapted from the paper *"Fritz Zwicky, Morphologie and Policy Analysis,"* 16th EURO Conference on Opertional Analysis, Brussels, 1998

[87] Lövkvist-Andersen, A-L., Olsson, R., Ritchey, T., and Stenström, M., "Modelling Society's Capacity to Manage Extraordinary Events - Developing a Generic Design Basis (GDB) Model for Extraordinary Societal Events using Computer-Aided Morphological Analysis," *adapted from a paper presented at the SRA (Society for Risk Analysis), Paris*, November 2004

[88] Isaksson, S., and Ritchey, T., "Protection against Sabotage of Nuclear Facilities: Using Morphological Analysis in Revising the Design Basis Threat," *adaptation of a Paper delivered to the 44th Annual Meeting of the Institute of Nuclear Materials Management, Phoenix, Arizona*, July 2003

[89] Ritchey, T., "Nuclear Facilities and Sabotage: Using Morphological Analysis as a Scenario and Strategy Development Laboratory," *adaptation of a Paper delivered to the 44th Annual Meeting of the Institute of Nuclear Materials Management, Phoenix, Arizona*, July 2003

[90] Ritchey, T., "Modeling Multi-Hazard Disaster Reduction Strategies with Computer-Aided Morphological Analysis," *3rd International ISCRAM Conference (B. Van de Walle and M. Turoff, eds.), Newark, NJ (USA)*, May 2006

[91] Ritchey, T., Stenström, M., and Eriksson, H., "Using Morphological Analysis for evaluating Preparedness for Accidents Involving Hazardous Materials," *adapted from an article presented at the 4th International Conference for Local Authorities, Shanghai*, November 2002

[92] Stults, I, and Jimenez, H., "Air Transportation System Critical Infrastructure Protection: A Morphological Approach to Policy Development," *Sam Nunn Security Program, Center for International Strategy, Technology and Policy, Sam Nunn School of International Affairs, Ivan Allen College of Liberal Arts, Georgia Institute of Technology*, November 2007

[93] Jimenez, H., Stults, I., and Mavris, D., "A Morphological Approach for Proactive Risk Management in Civil Aviation Security," *American Institute of Aeronautics and Astronautics, 47th AIAA Aerospace Sciences Meeting, Orlando, Florida*, January 2009

[94] Lovkvist-Andersen, A., Olsson, R., Ritchey, T., and Stenstrom, M., "Developing a Generic Design Basis Model for Extraordinary Societal Events using Computer-Aided Morphological Analysis," *Adapted from the Society for Risk Analysis Conference, Paris*, November 2004

[95] Zang, T. A., Hemsch, M. J., Hilburger, M. W., Kenny, S. P., Luckring, J., Maghami, P., Padula, S. L., and Stroud, W. J., "Needs and Opportunities for Uncertainty-Based

Multidisciplinary Design Methods for Aerospace Vehicles," *Technology Repository TM-2002-211462, NASA Langley Research Center, Hampton, VA*, 2002

[96] Forsberg, K., and Mooz, H., "The relationship of systems engineering to the project cycle," *1st Annual Symposium of the National Council On Systems Engineering (NCOSE), Chattanooga, TN*, October 1991

[97] Research and Innovative Technology Administration (RITA)

[98] Royce, W. W., "Managing the Development of Large Software Systems," *IEEE WESCON*, pp. 1-9, August 1970. Reprinted in Tutorial: *"Software Engineering Project Management,"* edited by R.H. Thayer, *IEEE Computer Society Press, Washington D.C.*, 1988

[99] Oberkampf, W. L., DeLand, S., Rutherford, B., Diegert, K., and Alvin, K., "Estimation of Total Uncertainty in Modeling and Simulation," *Sandia National Laboratories*, 2000

[100] Oberkampf, W. L., Helton, J. C., and Sentz, K., "Mathematical Representation of Uncertainty," *AIAA Non-Deterministic Approaches Forum, Seattle, WA*, April 2001

[101] Du, X., and Chen, W., "An Efficient Approach to Probabilistic Uncertainty in Simulation-based Multidisciplinary Design," *38th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada*, January 2000

[102] Bae, H-R., and Grandhi, R. V., "Uncertainty Quantification of Structural Response Using Evidence Theory," *AIAA – Journal*, Vol. 41, No. 10, October 2003

[103] Antonsson, E. K., and Otto, K. N., "Imprecision in Engineering Design," *ASME Journal of Mechanical Design*, Vol. 117(B), pp. 25-32, 1995

[104] Zhou, L., and Currie, K., "Uncertainty Analysis in the Preliminary Design Stage for a Robust Multidisciplinary Methodology," *10th AIAA/ISSMO Multidisciplinary Analysis and optimization Conference, Albany, New York*, August-September 2004

[105] Du, X., and Chen, W., "Efficient Uncertainty Analysis methods for Multidisciplinary Robust Design," *AIAA Journal*, Vol. 40, No. 3, March 2002

[106] Talley, D. N., and Mavris, D. N., "Robust Design Method for a Battlespace System of Systems," *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Shaumbourg*, IL, April 2008

[107] Dieter, G. E., "Engineering Design: A Materials and Processing Approach," *McGraw Hill*, 2000

[108] Huyse, L., "Solving Problems of Optimization Under Uncertainty as Statistical Decision Problems," *42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, April 2001

[109] Yen, J., and Langari, R., "Fuzzy Logic: Intelligence, Control, and Information" *Prentice Hall*, 1999

[110] Oberkampf, W. L., and Helton, J. C., "Investigation of Evidence Theory for Engineering Applications," *AIAA 4th Non-Deterministic Approaches Forum, Denver, CO*, April 2002

[111] Ben-Haim, Y., "Information Gap - Decision Theory - Decisions under severe uncertainty," *Academic Press, new York*, 2001

[112] Encyclopedia Britannica: definition of *"Probability Theory"*

[113] Thesaurus Dictionary: definition of *"Probabilistic"*

[114] O'Connor, J. J., and Robertson, E. F., "The MacTutor History of Mathematics archive: Pierre de Fermat," *University of St Andrews, School of Mathematics and Statistics, Scotland*, December 1996

[115] Christiaan Huygens, Letter to Pierre Perrault, "Sur la préface de M. Perrault de son traité de l'Origine des fontaines," [1763], *Oeuvres Complètes de Christiaan Huygens* (1897), Vol. **7**, pp. 298, quoted in Jacques Roger, *"The Life Sciences in Eighteenth-Century French Thought,"* Edition Keith R. Benson and translation Robert Ellrich, 1997

[116] Shafer, G., and Vovk , V., "The origins and legacy of Kolmogorov's Grundbegriffe," extract from *"The Game-Theoretic Probability and Finance Project,"* working paper # 4, October 2005

[117] Mavris, D. N., DeLaurentis, D. A., Bandte, O., and Hale, M. A., "A Stochastic Approach to Multi-disciplinary Aircraft Analysis and Design," *AIAA 36th Aerospace Sciences Meeting and Exhibit, Reno, NV*, January 1998

[118] Chen, W., Jin, R., and Sudjianto, A., "Analytical Uncertainty Propagation via Metamodels in Simulation-based Design under Uncertainty," *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, New York*, August-September 2004

[119] DeLaurentis, L. A., and Mavris, D. N., "Uncertainty Modeling and Management in Multidisciplinary Analysis and Synthesis," *AIAA 38th Aerospace Sciences Meeting & Exhibit, Reno, NV*, January 2000

[120] Pate-Cornell, M. E., “Uncertainties in risk analysis: Six levels of treatment,” *Reliability Engineering and System Safety*, Vol. 54, 1996

[121] Shonkwiler, R., “Math 4225 Class Notes,” *School of Mathematics, Georgia Institute of Technology*, 2004

[122] Fox, E. P., “The Pratt & Whitney Probabilistic Design System,” *35th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Hilton Head, SC*, 1994

[123] Mavris, D.N., and Bandte, O., “Comparison of Two Probabilistic Techniques for the Assessment of Economic Uncertainty,” *19th Annual Conference of the International Society of Parametric Estimators, New Orleans, LA*, May 1997

[124] Bandte, O., Mavris, D.N., and DeLaurentis, D.A., “Viable Design Through a Joint Probabilistic Estimation Technique,” *4th World Aviation Congress and Exposition, San Francisco, CA*, October 1999

[125] Bandte, O., “A Probabilistic Multi-Criteria Decision Making Technique for Conceptual and Preliminary Aerospace Systems Design,” *PhD thesis, Georgia Institute of Technology*, 2001

[126] Bandte, O., Mavris, D.N., and DeLaurentis, D.A., “Viable Designs Through a Joint Probabilistic Estimation Technique.” *AIAA/SAE World Aviation Congress*, October 1999

[127] Thesaurus Dictionary: definition of *“Modeling”*

[128] Gajda, W. J., and Biles, W. C., “Engineering Modeling and Computation,” *Boston: Houghton Mifflin Company*, 1978

[129] Przemieniecki, J., “Mathematical Methods in Defense Analysis,” *AIAA, Third Edition*, 2000

[130] Kapper, F. B., “Wargaming,” *Department of Defense*, 1981

[131] Franklin, W. R., Ray, C. K., and Mehta, S., “Geometric Algorithms for Siting of Air Defense Missile Batteries,” *Research Project for Batelle, Contract Number DAAL-86-D-0001, Electrical, Computer and Systems Engineering Department, Rensselaer Polytechnic Institute, Troy, NY*, 1994

[132] Kewley, R. H., and Embrechts, M. J., “Fuzzy-Genetic Decision Optimization for Positioning of Military Combat Units,” *IEEE International Conference on Systems, Man, and Cybernetics, La Jolla, CA*, pp. 3658-3664, 1998

[133] Can, T., Isler, V., and Ipekkan, Z., "Sensor Optimization in a Virtual Environment," *9th Conference on Computer Generated Forces and Behavioral Representation, Orlando, FL*, pp. 421-426, 2000

[134] Vasquez, M., and Hao, J. K., "A Heuristic Approach for Antenna Positioning in Cellular Networks," *Journal of Heuristics*, Vol. 7, Issue 5, pp. 443-472, 2001. doi:10.1023/A:1011373828276

[135] Kim, Y. H., and Clarke, G., "Exploring Optimal Visibility Site Selection Using Spatial Optimisation Techniques: The Potential of Geographical Information Science," *GISRUK'2001, University of Glamorgan, Wales*, pp. 141-146, April 2001

[136] Dhillon, S. S., and Chakrabarty, K., "Sensor Placement for Effective Coverage and Surveillance in Distributed Sensor Networks," *IEEE*, pp. 1609-1614, 2003

[137] Murray, A. T., Kim, K., Davis, J. W., Machiraju, R., and Parent, R., "Coverage optimization to support security monitoring," *Computers, Environment and Urban Systems*, Vol. 31, pp. 133-147, 2007. doi:10.1016/j.compenvurbsys.2006.06.002

[138] Krause, A., Leskovec, J., Guestrin, C., VanBriensen, J., ASCE, M., and Faloutsos, C., "Efficient Sensor Placement Optimization for Securing Large Water Distribution Networks," *Journal of Water Resources Planning and Management*, pp. 516-526, November-December 2008. doi: 10.1061/(ASCE)0733-9496(2008)134:6(516)

[139] Bottino, A., and Laurentini, A., "A nearly optimal sensor placement algorithm for boundary coverage," *Pattern Recognition*, Vol. 41, pp. 3343-3355, May 2008. doi:10.1016/j.patcog.2008.05.001

[140] Hsieh, Y.-C., Lee, Y.-C., and Chen, T.-C., "An immune based two-phase approach for the multiple-type surveillance camera location problem," *Expert Systems with Applications*, Vol. 36, pp. 10634-10639, 2009. doi:10.1016/j.eswa.2009.02.077

[141] Hamel, D., Chwatek, M., Garcia, S., Farouk, B., Kam, M., and Dandekar, K. R., "Sensor Placement for Urban Homeland Security Applications," *International Journal of Distributed Sensor Networks*, Vol. 2012, 15 pages, October 2010. doi:10.1155/2010/859263

[142] Xu, Y.-C., Lei, B., and Hendriks, E. A., "Camera Network Coverage Improving by Particle Swarm Optimization," *EURASIP Journal on Image and Video Processing*, Vol. 2011, 10 pages, November 2010. doi:10.1155/2011/458283

[143] Tao, D., Tang, S., Zhang, H., Mao, X., and Ma, H., "Strong barrier coverage in directional sensor networks," *Computer Communications*, Vol. 35, pp. 895-905, February 2012. doi:10.1016/j.comcom.2012.01.022

[144] Liao, W.-H., Kao, Y., and Li, Y.-S., “A sensor deployment approach using glowworm swarm optimization algorithm in wireless sensor networks,” *Expert Systems with Applications*, Vol. 38, pp. 12180-12188, 2011. doi:10.1016/j.eswa.2011.03.053

[145] Liao, W.-H., Kao Y., and Wu, R.-T., “Ant colony optimization based on sensor deployment protocol for wireless sensor networks,” *Expert Systems with Applications*, Vol. 38, pp. 6599-6605, 2011. doi:10.1016/j.eswa.2010.11.079

[146] Pradhan P. M., and Panda, G., “Connectivity constrained wireless sensor deployment using multiobjective evolutionary algorithms and fuzzy decision making,” *Ad Hoc Networks*, Vol. 10, pp. 1134-1145, March 2012. doi:10.1016/j.adhoc.2012.03.001

[147] Guerriero F., Violi, A., Natalizio, E., Loscri, V., and Costanzo, C., “Modelling and solving optimal placement problems in wireless sensor networks,” *Applied Mathematical Modeling*, Vol. 35, pp. 230-241, June 2010. doi:10.1016/j.apm.2010.05.020

[148] Prommak, C., and Modhirun, S., “Optimal Network Design for Efficient Energy Utilization in Wireless Sensor Networks,” *Journal of Computer Science*, Vol. 8, No. 1, pp. 149-158, 2012. ISSN 1549-3636

[149] Chaudhry, S. B., Hung, V. C., Guha, R. K., and Stanley, K. O., “Pareto-based evolutionary computational approach for wireless sensor placement,” *Engineering Applications of Artificial Intelligence*, Vol. 24, pp. 409-425, 2011. doi:10.1016/j.engappai.2010.07.007

[150] Younis, M., and Akkaya, K., “Strategies and techniques for node placement in wireless sensor networks: A survey,” *Ad Hoc Networks*, Vol. 6, pp. 621-655, 2008. doi:10.1016/j.adhoc.2007.05.003

[151] Guvensan, M. A., and Yavuz, A. G., “On coverage issues in directional sensor networks: A survey,” *Ad Hoc Networks*, Vol. 9, pp. 1238-1255, February 2011. doi:10.1016/j.adhoc.2011.02.003

[152] Erdem, U. M., and Sclaroff, S., “Optimal Placement of Cameras in Floorplans To Satisfy Task Requirements and Cost Constraints,” *International Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras*, pp. 30-41, 2003

[153] De Jong, K., “An analysis of the behaviour of a class of genetic adaptive systems,” *PhD thesis, Department of Computer Science, University of Michigan, Ann Harbor, MI*, 1975

[154] Dhillon, S. S., Chakrabarty, K., and Iyengar, S. S., “Sensor Placement for Grid Coverage under Imprecise Detections,” *DARPA grant number grant number N66001-00-1-8946, Office of Naval Research grant number N00014-01-1-0712*, 2002

[155] Telci, I. T., Nam, K., Guan, J., and Aral, M.M., “Optimal water quality monitoring network design for river systems,” *Journal of Environmental Management*, Vol. 90, pp. 2987-2998, June 2009. doi:10.1016/j.jenvman.2009.04.011

[156] Shen, H., and McBean, E., “Pareto Optimality for Sensor Placements in a Water Distribution System,” *Journal of Water Resources Planning and Management ASCE*, pp. 243-248, May-June 2011. doi:10.1061/(ASCE)WR.1943-5452.0000111

[157] Xu, J., Johnson, M. P., Fischbeck, P. S., Small, M. J, and VanBriesen, J. M., “Robust placement of sensors in dynamic water distribution systems,” *European Journal of Operational Research*, Vol. 202, pp. 707-716, 2010. doi:10.1016/j.ejor.2009.06.010

[158] Isovitsch, S. L., and VanBriesen, J. M., “Sensor Placement and Optimization Criteria Dependencies in a Water Distribution System,” *Journal of Water Resources Planning and Management ASCE*, pp. 186-196, March-April 2008. doi:10.1061/ASCE/0733-9496/2008/134:2/186

[159] Trujillo-Ventura, A., and Ellis, J. H., “Multiobjective Air Pollution Monitoring Network Design,” *Atmospheric Environment*, Vol. 25A, No. 2, pp. 469-479, 1991

[160] Abdalla, M., and El-Khawaldeh, E., “Optimal Damage Detection Sensor Placement using PSO,” *Applied Mechanics and Materials*, Vols. 110-116, pp. 5336-5341, 2012. doi:10.4028/www.scientific.net/AMM.110-116.5336

[161] Yan, Y., Chen, H., and Jiang, J., “Optimal Placement of Sensors for Damage Characterization using Genetic Algorithms,” *Key Engineering Materials*, Vols. 334-335, pp. 1033-1036, March 2007. doi:10.4028/www.scientific.net/KEM.334-335.1033

[162] Worden, K., and Burrows, A. P., “Optimal sensor placement for fault detection,” *Engineering Structures*, Vol. 23, 2001, pp. 885-901. PII: S0141-0296(00)00118-8

[163] Minjie, D., Jinyan, C., Limin, L., and Peng, C., “ARRs based sensor placement optimization for fault diagnosis,” *Procedia Engineering*, Vol. 16, 2011, pp. 42-47

[164] Zhan, S., Huang, F., and Lai, J., “Optimal sensor placement for high-rise building via genetic algorithms and improved information matrix criterion,” *Advanced Materials Research*, Vols. 368-373, pp. 1653-1659, 2012. doi:10.4028/www.scientific.net/AMR.368-373.1653

[165] Yang, J., and Zhang, L., “Optimization of Sensor Placement in SHM Based on the Dual Coded Genetic Algorithm,” *Applied Mechanics and Materials*, Vol. 151, pp. 139-144, 2012. doi:10.4028/www.scientific.net/AMM.151.139

[166] Han-bing, L., Chun-li, W., and Jing, W., "Sensor Optimal Placement for Bridge Structure based on Single Parents Genetic Algorithm with Different Fitness Functions," *Key Engineering Materials*, Vol. 456, pp. 115-124, 2011. doi:10.4028/www.scientific.net/KEM.456.115

[167] Guratzsch, R. F., and Mahadevan, S., "Structural Health Monitoring Sensor Placement Optimization Under Uncertainty," *AIAA Journal*, Vol. 48, No. 17, pp. 1281-1289, July 2010. doi:10.2514/1.28435

[168] Franchi, C. G., and Gallieni, D., "Genetic Algorithm-Based Procedure for Pretest Analysis," *AIAA Technical Notes*, Vol. 33, No. 7, pp. 1362-1364, August 1994

[169] Boying, Q., and Xiankun, L., "Optimal Sensor Placement Based on Particle Swarm Optimization," *Advanced Materials Research*, Vols. 271-273, pp. 1108-1113, 2011. doi:10.4028/www.scientific.net/AMR.271-273.1108

[170] Xiao, R.–B., Xu, Y.–C., and Amos, M., "Two hybrid compaction algorithms for the layout optimization problem," *BioSystems*, Vol. 90, pp. 560-567, 2007. doi:10.1016/j.biosystems.2006.12.007

[171] Boying, Q., and Xiankun, L., "Application of Integer-coded Genetic Algorithm to Optimal Sensor Placement," *Advanced Materials Research*, Vols. 271-273, pp. 1114-1119, 2011. doi:10.4028/www.scientific.net/AMR.271-273.1114

[172] Kim, J., Park, B., Lee, J., and Won, J., "Determining optimal sensor locations in freeway using genetic algorithm-based optimization," *Engineering Applications of Artificial Intelligence*, Vol. 24, pp. 318-324, 2011. doi:10.1016/j.engappai.2010.10.020

[173] Li, J., Kapania, R. K., and Spillman Jr., W. B., "Placement Optimization of Distributed-Sensing Fiber-Optic Sensors Using Genetic Algorithms," *AIAA Journal*, Vol. 46, No. 4, pp. 824-836, April 2008

[174] Stanley, K. O., and Miikkulainen, R., "Competitive Coevolution through Evolutionary Complexification," *Journal of Artificial Intelligence Research*, Vol. 21, pp. 63-100, February 2004

[175] Molyboha, A., and Zabarankin, M., "Stochastic Optimization of Sensor Placement for Diver Detection," *Operations Research*, Vol. 60, No. 2, pp. 292-312, March-April 2012. doi:10.1287/opre.1110.1032

[176] Gao, G., and Zeng, X., "Sensor Location Optimization for Distributed Parameter Systems Based on Wavelet Transforms," *Advanced Materials Research*, Vols. 482-484, pp. 1747-1753, February 2012. doi:10.4028/www.scientific.net/AMR.482-484.1747

[177] Krohling, R. A., Knidel, H., and Shi, Y., “Solving numerical equations of hydraulic problems using PSO,” *IEEE CEC*, 2002

[178] Heo, N., and Varshney, P. K., “Energy–efficient deployment of intelligent mobile sensor network,” *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 35, pp. 78–92, 2005

[179] Kantorovich, L. V., “A new method of solving some classes of extremal problems,” Doklady Akademii Nauk SSSR, Vol. 28, pp. 211-214, 1940

[180] Dantzig, G. B., “Maximization of a linear function of variables subject to linear inequalities,” 1947, In *Activity Analysis of Production and Allocation*, Koopmans, T. C., New York-London, Wiley & Chapman-Hall , 1951

[181] Kuhn, H. W., and Tucker, A. W., “John von Neumann's work in the theory of games and mathematical economics,” *Bulletin of American Mathematical Society*, Vol. 64, No. 3, Part 2, pp. 100-122, 1958

[182] Dantzig, G. B., and Thapa, M. N., “Linear programming 1: Introduction,” *Springer-Verlag*, 1997

[183] Dantzig, G. B., and Thapa, M. N., “Linear Programming 2: Theory and Extensions,” *Springer-Verlag*, 2003

[184] Borgwardt, K.-H., “The Simplex Algorithm: A Probabilistic Analysis,” *Algorithms and Combinatorics*, Vol. 1, *Springer-Verlag*, 1987

[185] Boyd, S. P., and Vandenberghe, L., “Convex Optimization,” *Cambridge University Press*, 2004. ISBN 978-0-521-83378-3

[186] Boţ, R. I., Wanka, G., Grad, S.-M., “Duality in Vector Optimization,” *Springer*, 2009. ISBN 978-3-642-02885-4

[187] Zălinescu, C., “Convex analysis in general vector spaces,” *River Edge, NJ: World Scientific Publishing Co., Inc.*, pp. 106-113, 2002. ISBN 981-238-067-1

[188] Berkovitz, L. D., “Convexity and Optimization in $\mathbb{R}^n$,” *New York: John Wiley & Sons*, 2001. ISBN 978-0-471-35281-5

[189] Kutateladze, S., “The Farkas Lemma Revisited,” *Siberian Mathematical Journal*, Vol. 51, No. 1, pp. 78–87, 2010. doi:10.1007/s11202-010-0010-y

[190] Vazirani, V., “Approximation Algorithms,” *Springer-Verlag New York*, 2001, ISBM: 3-540-65367-8

[191] Donev, A., Stillinger, F., Chaikin, P., and Torquato, S., "Unusually Dense Crystal Packings of Ellipsoids," *Physical Review Letters*, Vol. 92., No. 25, 2004. doi:10.1103/PhysRevLett.92.255506

[192] Hudson, T. S., and Harrowell, P., "Structural searches using isopointal sets as generators: Densest packings for binary hard sphere mixtures," *Journal of Physics: Condensed Matter,* Vol. 23, No. 19, pp. 194103, 2011. doi:10.1088/0953-8984/23/19/194103

[193] Torquato, S., and Jiao, Y., "Dense packings of the Platonic and Archimedean solids," *Nature*, Vol. 460, No. 7257, pp. 876-879, 2009. doi:10.1038/nature08239

[194] Betke, U., and Henk, M., "Densest lattice packings of 3-polytopes," *Computational Geometry*, Vol. 16, pp. 157-186, 2000

[195] Haji-Akbari, A., Engel, M., Keys, A. S., Zheng, X., Petschek, R. G., Palffy-Muhoray, P., and Glotzer, S. C., "Disordered, quasicrystalline and crystalline phases of densely packed tetrahedra," *Nature*, Vol. 462, No. 7274, pp. 773-777, 2009. doi:10.1038/nature08641

[196] Chen, E. R., Engel, M., and Glotzer, S. C., "Dense Crystalline Dimer Packings of Regular Tetrahedra," *Discrete & Computational Geometry*, Vol. 44, No. 2, pp. 253, 2010. doi:10.1007/s00454-010-9273-0

[197] Croft, H. T., Falconer, K. J., and Guy, R. K., "Unsolved Problems in Geometry", *Springer-Verlag New York*, pp. 108-110, 1991. ISBN 0-387-97506-3

[198] Specht, E., "The best known packings of equal circles in a square," retrieved from its website

[199] Specht, E., "The best known packings of equal circles in an isosceles right triangle," retrieved from its website

[200] Melissen, J., "Packing 16, 17 or 18 circles in an equilateral triangle," *Discrete Mathematics*, Vol. 145, pp. 333-342, 1995. doi:10.1016/0012-365X(95)90139-C

[201] Friedman, E., "Packing unit squares in squares: a survey and new results," *The Electronic Journal of Combinatorics,* Vol. DS7, 2005

[202] Erdős, P., and Graham, R. L., "On packing squares with equal squares," *Journal of Combinatorial Theory, Series A*, Vol. 19, pp. 119-123, 1975

[203] Birgin, E. G., Lobato, R. D., and Morabito, R., "An effective recursive partitioning approach for the packing of identical rectangles in a rectangle," *Journal of the Operational Research Society*, Vol. 61, pp. 306-320, 2010

[204] Gyetvan, F., and Shi, Y., “Weak duality theorem and complementary slackness theorem for linear matrix programming problems,” *Operations Research*, Vol. 11, No. 4, pp. 249-253, 1992

[205] Berenstein, C. A., and Gay, R., “Complex Variables: An Introduction,” *Springer-Graduate Texts in Mathematics*, 1997. ISBN 0-387-97349-4

[206] Rockafellar, R. T., “Convex analysis,” *Princeton: Princeton University Press*, 1970

[207] Kozlov, M. K., Tarasov, S. P., and Khachiyan, L. G., “The polynomial solvability of convex quadratic programming,” *USSR Computational Mathematics and Mathematical Physics*, Vol. 20, No. 5, pp. 223-228, 1980

[208] Karmarkar, N., “A new polynomial-time algorithm for linear programming,” *Proceeding STOC’84 of the sixteenth annual ACM symposium on Theory of Computing*, pp. 302-311, 1984

[209] Padberg, M., “Linear Optimization and Extensions,” *Second Edition, Springer-Verlag*, 1999

[210] Bland, R. G., “New finite pivoting rules for the simplex method”, *Mathematics of Operations Research*, Vol. 2, pp. 103-107, 1977

[211] Papadimitriou, C. H., and Steiglitz, K., “Combinatorial Optimization: Algorithms and Complexity,” *Courier Dover Publications*, 1998

[212] Serang, O., “Conic Sampling: An Efficient Method for Solving Linear and Quadratic Programming by Randomly Linking Constraints within the Interior,’ *PLOS ONE* **7**, Vol. 8, 2012. doi:10.1371/journal.pone.0043706

[213] Fukuda, K., Terlaky, T., “Criss-cross methods: A fresh view on pivot algorithms,” *Mathematical Programming: Series B, 16th International Symposium on Mathematical Programming held in Lausanne*, Vol. 79, pp 369-395, 1997. doi:10.1016/S0025-5610(97)00062-2

[214] Roos, C., “An exponential example for Terlaky's pivoting rule for the criss-cross simplex method,” *Mathematical Programming Series A*, Vol. 46, No. 1, pp.79, 1990. doi:10.1007/BF01585729

[215] Fukuda, K., Namiki, M., “On extremal behaviors of Murty's least index method,” *Mathematical Programming*, Vol. 64, No. 1, pp. 365-370, 1994. doi:10.1007/BF01582581

[216] Fukuda, K., Terlaky, T., “Criss-cross methods: A fresh view on pivot algorithms,” *Mathematical Programming: Series B, 16th International Symposium on*

Mathematical Programming held in Lausanne , Vol. 79, pp. 369-395, 1997. doi:10.1016/S0025-5610(97)00062-2

[217] Gondzio, J., Terlaky, T., “A computational view of interior point methods,” In *Advances in linear and integer programming, Beasley, J. E., Oxford Lecture Series in Mathematics and its Applications*, Vol. 4, pp. 103-144, 1996

[218] Chandru, V., and Rao, M. R., “Linear Programming,” Chapter 31 and 32 in *Algorithms and Theory of Computation Handbook*, edited by Atallah, M. J., CRC Press, 31-1 to 31-37, 32-1 to 32-45, 1999

[219] Strang, G., “Karmarkar's algorithm and its place in applied mathematics,” *The Mathematical Intelligencer*, *New York: Springer*, Vol. 9, No. 2, pp. 4-10, 1987. doi:10.1007/BF03025891

[220] Andersen, K., Louveaux, Q., and Weismantel, R., “An Analysis of Mixed Integer Linear Sets Based on Lattice Point Free Convex Sets,” *Mathematics of Operations Research*, Vol. 35, No. 1, pp. 233-256, 2010

[221] Vielma, J. P., Ahmed, S., and Nemhauser, G., “Mixed-Integer Models for Nonspearable Piecewise-Linear Optimization: Unifying Framework and Extensions,” *Operations Research*, Vol. 58, No. 2, pp. 303-315, 2010

[222] Garey, M. R., and Johnson, D. S., “Computers and Intractability: A Guide to the Theory of NP-Completeness,” *W. H. Freeman*, 1979. ISBN 0-7167-1045-5

[223] Karp, R. M., “Reducibility Among Combinatorial Problems,” In *Complexity of Computer Computations*, Miller, R. E., and Thatcher, J. W., New York: Plenum, pp. 85-103, 1972

[224] Giles, F.R. and Pulleyblank, W. R., “Total Dual Integrality and Integer Polyhedra,” *Linear algebra and its applications*, Vol. 25, pp. 191-196, 1979

[225] Cornuejols, G., “Valid Inequalities for Mixed Integer Linear Programs,” *Mathematical Programming Series B*, Vol. 112, pp. 3-44, 2008

[226] Land, A. H., and Doig, A. G., “An automatic method of solving discrete programming problems,” *Econometrica*, Vol. 28, No. 3, pp. 497-520, 1960. doi:10.2307/1910129

[227] Achterberg, T., Koch, T., and Martin, A., “Branching rules revisited,” *Operations Research*, Vol. 33, No. 1, pp. 42-54, 2005

[228] Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P., and Vance, P. H., “Branch-and-price: column generation for solving huge integer programs,” *Operations Research*, Vol. 46, No. 3, pp. 316-329, 1998

[229] Feillet, D., “A tutorial on column generation and branch-and-price for vehicle routing problems,” *Operation Research*, Vol. 8, No. 4, pp. 407-424, 2010

[230] Burkov, A., and Chaib-draa, B., “An Approximate Subgame-Perfect Equilibrium Computation Technique for Repeated Games,” *Twenty-Fourth AAAI Conference on Artificial Intelligence*, pp. 729-736, 2010

[231] Jones, J. G., Meneghin, B. J., and Kirby, M. W., “Formulating Adjacency Constraints in Linear Optimization Models for Scheduling Projects in Tactical Planning,” *Forest Science*, Vol. 37, No. 5, pp. 1283-1297, 1991

[232] Farag, A., Al-Baiyat, S., and Cheng, T. C., “Economic load dispatch multiobjective optimization procedures using linear programming techniques,” *IEEE Transactions on Power Systems*, Vol. 10, No. 2, pp. 731-738, 1995

[233] Ho, J. K., “A Successive Linear Optimization Approach to the Dynamic Traffic Assignment Problem,” *Transportation Science*, Vol. 14, No. 4, pp. 295-305, 1980

[234] Tanczos, K., and Torok, A., “Linear optimization model of urban areas’ operating efficiency,” *Transport*, Vol. 22, No. 3, pp. 225-228, 2007. doi: 10.1080/16484142.2007.9638129

[235] Nagel, J., “Determination of an economic energy supply structure based on biomass using mixed-integer linear optimization model,” *Ecological Engineering*, Vol. 16, supp. 1, pp. 91-102, 2000

[236] Van den Broek, M., Brederode, E., Ramirez, A., Kramers, L., Van der Kuip, M., Wildenborg, T., Turkenburg, W., and Faaij, A., “Designing a cost-effective CO_2 storage infrastructure using a GIS based linear optimization energy model,” *Environmental Modelling & Software*, Vol. 25, No. 12, pp. 1754-1768, 2010

[237] Robertazzi, T. G., “Planning Telecommunication Networks,” *First Edition, Wiley-IEEE Press*, 1998. ISBN: 0780347021

[238] Hung, Y.-F., and Leachman, R. C., “A production planning methodology for semiconductor manufacturing based on iterative simulation and linear programming calculations,” *IEEE Transactions on Semiconductor Manufacturing*, Vol. 9, No. 2, pp. 257-269, 1996

[239] Poh, K. L., and Quaddus, M. A., “A hybrid approach to multi-objective linear optimization,” *Journal of the Operational Research Society*, Vol. 41, No. 11, pp. 1037-1048, 1990

[240] Vanderbei, R. J., “Linear Programming: Foundations and Extensions,” *Third Edition, International Series in Operations Research & Management Science*, Vol. 114, *Springer Verlag*, 2008. ISBN 978-0-387-74387-5

[241] Bazaraa, M. S., Jarvis, J. J., and Sherali, H. D., "Linear Programming and Network Flows," *Fourth Edition, John Wiley & Sons, Inc., Hoboken, NJ, USA*, 2011. doi: 10.1002/9780471703778.fmatter

[242] Lee, H., Pinto, J. M., Grossmann, I. E., and Park, S., "Mixed-Integer Linear Programming Model for Refinery Short-Term Scheduling of Crude Oil Unloading with Inventory Management," *Industrial & Engineering Chemistry Research*, vol. 35, No. 5, pp. 1630-1641, 1996. doi: 10.1021/ie950519h

[243] Ramanathan, R., "ABC inventory classification with multiple-criteria using weighted linear optimization," *Computers & Operations Research*, Vol. 33, No. 3, pp. 695-700, 2006

[244] Spall, J. C., "Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control," *Wiley-Interscience Series in Discrete Mathematics and Optimization*, 2003. ISBN: 978-0-471-33052-3

[245] Fu, M. C., "Optimization for Simulation: Theory vs. Practice," *INFORMS Journal on Computing*, Vol. 14, No. 3, pp. 192-227, 2002. doi:10.1287/ijoc.14.3.192.113

[246] Robbins, H., and Monro, S., "A Stochastic Approximation Method," *Annals of Mathematical Statistics*, Vol. 22, No. 3, pp. 400-407, 1951. doi:10.1214/aoms/1177729586

[247] Kiefer, J., and Wolfowitz, J., "Stochastic Estimation of the Maximum of a Regression Function," *Annals of Mathematical Statistics*, Vol. 23, No. 3, pp. 462-466, 1952. doi:10.1214/aoms/1177729392

[248] Hu, C., Kwok, J., and Pan, W., "Accelerated gradient methods for stochastic optimization and online learning," *In NIPS*, 2009

[249] Spall, J. C., "Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation," *IEEE Transactions on Automatic Control*, Vol. 37, No. 3, pp. 332-341, 1992. doi:10.1109/9.119632

[250] Hoos, H. H., and Stützle, T., "Stochastic Local Search: Foundations and Applications," *Morgan Kaufmann - Elsevier*, 2004

[251] Wolpert, D. H., Bieniawski, S. R., and Rajnarayan, D. G., "Probability Collectives in Optimization," *Rao, C. R., Govindaraju, V. Editions*, 2011

[252] Battiti, R., and Gianpietro, T., "The reactive tabu search," *ORSA Journal on Computing*, Vol. 6, No. 2, pp. 126-140, 1994

[253] Battiti, R., Brunato, M., and Mascia, F., “Reactive Search and Intelligent Optimization,” *Springer Verlag*, 2008. ISBN 978-0-387-09623-0

[254] Zhigljavsky, A. A., “Theory of Global Random Search,” Kluwer Academic, 1991. ISBN 0-7923-1122-1

[255] Spall, J. C., “Handbook of computational Statistics,” *Gentle, J., Härdle, W., and Mori, Y. editions, Springer Heidelberg*, 2004

[256] Wenzel, W., and Hamacher, K., “Stochastic tunneling approach for global optimization of complex potential energy landscapes,” *Physical Review Letters*, Vol. 82, No. 15, pp. 3003, 1999. doi:10.1103/PhysRevLett.82.3003

[257] Fouskakis, D., and Draper, D., “Stochastic Optimization: a Review,” *International Statistical Review*, Vol. 70, No. 3, pp. 315-349, 2002

[258] Solis, F. J., and Wets, J. B., “Minimization by Random Search Techniques,” *Mathematics and Operations Research*, Vol. 6, pp. 19-30

[259] Michalewicz, Z., and Fogel, D. B., “How to Solve It: Modern Heuristics,” *Springer-Verlag, New York*, 2000

[260] Spall, J. C., “Introduction to Stochastic search and Optimization: estimation, Simulation, and Control,” *Wiley, Hoboken, NJ*, Sections 10.6 and 12.2, 2003

[261] Rechenberg, I., “Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution,” *Frommann-Holzboog Verlag*, 1973

[262] Grimaccia, F., Mussetta, M., Pirinoli, P., and Zich, R. E., “Genetical Swarm Optimization (GSO): a class of Population-based Algorithms for Antenna Design,” *IEEE First*” *International Conference on Communications and Electronics*, pp. 467-471, 2006

[263] Holland, J., “Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence,” *Ann Arbor, The University of Michigan Press*, 1975

[264] Rennard, J-P., “Vie artificielle : Ou la biologie rencontre l’informatique (Illustré avec Java,” *Artificial Life : Where Biology Meets Computer Science (Illustrated in Java), Vuibert*, 2002

[265] Genetic Algorithm Warehouse, Artificial Intelligence Department, “Genetic algorithms”

[266] Mitchell, M., “An Introduction to Genetic Algorithms,” *MIT Press paperback edition, 1998, Massachusetts Institute of Technology*, 1996

[267] Man, K. F., Tang, K. S., and Kwong, S., "Genetic Algorithms: concepts and design," *Advanced Textbooks in control and signal processing, Springer-Verlag London*, 1999

[268] Haupt, R. L., and Haupt, S. E., "Practical Genetic Algorithms," *Second Edition, Willey-Interscience, A John Wiley & Sons, Inc., Publication*, 2004

[269] Volovoi, V., "Advanced Design Methods II Class Notes," *School of Aerospace Engineering, Georgia Institute of Technology*, 2007

[270] Eskandari, H., Rabelo, L., and Mollaghasemi, M., "Multiobjective Simulation Optimization Using an Enhanced Genetic Algorithm," *Winter Simulation Conference M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J.A. Joines Editions*, 2005

[271] Yi, W., Liu, Q., and He, Y., "Dynamic Distributed Genetic Algorithms," *4th International Conference on Genetic Algorithms, R. K. Belew, and L. B. Booler Editions*, 1991

[272] Sivanandam, S. N., and Deepa, S. N., "Introduction to Genetic Algorithms," *Springer-Verlag Berlin Heidelberg*, 2008

[273] Arabas, J., Michalewicz, Z., and Mulawka, J., "GAVaPS – a Genetic Algorithm with Varying Population Size," *1st IEEE Conference on Evolutionary Computation ICEC*, 1994

[274] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P., "Optimization by Simulated Annealing," *Science, New Series*, Vol. 220, pp. 671–680, 1983

[275] Cerny, V., "A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm," *Journal of Optimization Theory and Applications*, Vol. 45, pp. 41-51, 1985

[276] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E., "Equation of State Calculations by Fast Computing Machines," *Journal of Chemistry and Physics*, Vol. 21, pp. 1087, 1953

[277] Van Laarhoven, P. J. M., and Aarts, E. H. L., "Simulated Annealing: Theory and Applications," *Mathematics and Its Applications, Kluwer Academic Publishers*, 1988

[278] Kolar, D., Puksec, J. D., and Branica, I., "VLSI circuit partition using simulated annealing algorithm," *Proceedings of the 12th IEEE Mediterranean Electrotechnical Conference, MELECON*, Vol. 1, pp. 205-208, 2004

[279] Sandeep, S. G., Rajeevan, C., and Ashwani, K. C., "Efficient Clustering and Simulated Annealing Approach for Circuit Partitioning," *Journal of Shangai Jiaotong*

University Science, Vol. 16, No. 6, pp. 708-712, 2011. doi: 10.1007/s12204-011-1138-z

[280] Gill, S. S., Aneja, B., Chandel, R., and Chandel, A., "Simulated annealing based VLSI circuit partitioning for delay minimization," *Proceedings of the 4th WSEAS international conference on Computational Intelligence*, pp. 60-63, 2010

[281] Gill, S. S., Chandel, R., Chandel, A., and Parvinder, S. S., "Simulated Annealing based Delay Centric VLSI Circuit Partitioning," *Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT)*, Vol. 1, pp. 1-4, 2010

[282] Pattanick, S., Bhoi, S. P., and Nohanty, R., "Simulated Annealing Based Placement Algorithms and Research Challenges: A Survey," *Journal of Global Research in Computer Science*, Vol. 3, No. 6, pp. 33-37, 2012

[283] Van Laarhoven, P. J. M., Aarts, E. H. L., and Lenstra, J. K., "Job Scheduling by Simulated Annealing," *Operations Research*, Vol. 40, No. 1, pp. 113-125, 1992

[284] Yamada, T., Rosen, B. E., and Nakano, R., "A simulated annealing approach to job scheduling using critical block transition operators," Proceedings of the IEEE International Conference on Neural Networks, IEEE World Congress on Computational Intelligence, Vol. 7, pp.4687-4692, 1994

[285] Farmer, A., Smith, J. S., and Miller, L. T., "Scheduling Umpire Crews for Professional Tennis Tournaments," *Interfaces INFORMS*, Vol. 37, No. 2, pp. 187-196, 2007

[286] Trick, M. A., Yildiz, H., and Yunes, T., "Scheduling Major League Baseball Umpires and the Traveling Umpire Problem," *Interfaces INFORMS*, Vol. 42, No. 3, pp. 232-244, 2012

[287] Mastorocostas, P. A., Varsamis, D. N., and Mastorocostas, C. A., "A simulated annealing-based learning algorithm for block-diagonal recurrent neural networks," *Proceedings of the 24th IASTED International Conference on Artificial Intelligence and Applications*, pp. 244-249, 2006

[288] Lopez, A. M., "Simulated Annealing Based Hierarchical Q-Routing: A Dynamic Routing Protocol," *Master of Science Degree Paper, University of Science and Arts of Oklahoma, Chickasha, OK*, 2007

[289] Carnevali, P., Coletti, L., and Patarnello, S., "Image processing by simulated annealing," *IBM Journal of Research and Development*, Vol. 29, No. 6, pp. 569-579, 1985

[290] Robini, M. C., Rastello, T., and Magnin, I. E., "Simulated Annealing, Acceleration Techniques, and Image Restauration," *IEEE Transactions on Image Processing*, Vol. 8, No. 10, pp. 1374-1387, 1999

[291] Aguilera, C. A., Ramos, M. A., and Sappa, A. D., "Simulated Annealing: A Novel Application of Image Processing in the Wood Area," *Chapter 5 of "Simulated Annealing – Advances, Applications and Hybridizations," Marcos de Sales Guerra Tsuzuki Editions*, 2012. ISBN: 978-953-51-0710-1

[292] Aarts, E. H. L., and Korst, J., "Simulated annealing and Boltzmann machines: a stochastic approach to combinatorial optimization and neural computing," *Wiley-Interscience Series in Discrete Mathematics and Optimization, Advisory Editors Graham, R. L., Lenstra, J. K., Tarjan, R. E., "Discrete Mathematics and Optimization involves the study of finite structures,"* 1989

[293] Xu, L., and Oja, E., "Improved simulated annealing, Boltzmann machine, and attributed graph matching," *Neural Networks, Lecture Notes in Computer Science*, Vol. 412, pp. 151-160, 1990

[294] Jerrum, M., and Sorkin, G. B., "Simulated Annealing for Graph Bisection," *IEEE Symposium on Foundations of Computer Science*, pp. 94-103, 1993

[295] Johnson, D. A., Aragon, C. R., McGeoch, L. A., and Schevon, C., "Optimization by Simulated Annealing: An Experimental Evaluation; Part I, Graph Partitioning," *Operations Research*, Vol. 37, No. 6, pp. 865-892, 1989

[296] Nolte, A., and Schrader, R., "Simulated Annealing and Graph Coloring," *Combinatorics, Probability and Computing*, Vol. 10, No. 1, pp. 29-40, 2001

[297] Ayanegui, H., and Chavez-Aragon, A., "A complete algorithm to solve graph-coloring problem," *Proceedings of the Fifth Latin American Workshop on Non-Monotonic Reasoning LANMR*, Vol. 533, pp. 107-117, 2009

[298] Johnson, D. A., Aragon, C. R., McGeoch, L. A., and Schevon, C., "Optimization by Simulated Annealing: An Experimental Evaluation; Part II, Graph Coloring and Number Partitioning," *Operations Research*, Vol. 39, No. 3, pp. 378-406, 1991

[299] Ruml, W., Ngo, J. T., Marks, J., and Shieber, S. M., "Easily searched encodings for number partitioning," *Journal of Optimization Theory and Applications*, Vol. 89, No. 2, pp. 251-291, 1996

[300] Li, Y., Zhou, A., and Zhang, G., "Simulated annealing with probabilistic neighborhood for traveling salesman problems," *Proceedings of the Seventh International Conference on Natural Computation (ICNC)*, Vol. 3, pp. 1565-1569, 2011

[301] Lemes, M. R., Zacharias, C. R., and Pino, A. D. Jr., "Generalized simulated annealing: Application to silicon clusters," *American Physical Society, Physical Review B*, Vol. 55, No. 15, pp. 9279-9281, 1997

[302] Dolan, W. B., Cummings, P. T., and LeVan, M. D., "Process Optimization via Simulated Annealing: Application to Network Design," *AIChE Journal*, Vol. 35, No. 5, pp. 725-736, 1989

[303] Kaczmarski, K., and Antos, D., "Use of simulated annealing for optimization of chromatographic separations," *Acta Chromatographica*, Vol. 17, pp. 20-45, 2006

[304] Elmohamed, S., Fox, G., and Coddington, P., "A Comparison of Annealing Techniques for Academic Course Scheduling," *2nd International Conference on the Practice and Theory of Automated Timetabling*, pp. 146-166, 1998

[305] Bohachevsky, I., Johnson, M. E., and Stein, M. L., "Generalized simulated annealing for function optimization," *Technometrics*, Vol. 28, pp. 209-217, 1986

[306] Laud, P., Berliner, L. M., and Goel, P. K., "A stochastic probing algorithm for global optimization," *Computer Science and Statistics, 21st Symposium on the Interface*, pp. 220-229, 1989

[307] Van Laarhoven, P., Boender, C., Aarts, E., and Rinnooy Kan, A., "A Bayesian approach to simulated annealing," *Probability in the Engineering and Informational Sciences*, Vol. 3, pp. 453-475, 1989

[308] Glover, F., and McMillan, C., "The general employee scheduling problem. An integration of MS and AI," *Computers & Operations Research*, Vol. 13, No. 5, pp. 563-573, 1986

[309] Glover, F., "Tabu Search - Part 1," *ORSA Journal on Computing*, Vol. 1, No. 2, pp. 190-206, 1989

[310] Glover, F., "Tabu Search - Part 2," *ORSA Journal on Computing*, Vol. 2, No. 1, pp. 4-32, 1990

[311] Glover, F., "Tabu Search: A Tutorial," *Interfaces*, 1990

[312] Glover, F., and Laguna, M., "Tabu Search", *Dordrecht, Kluwer*, 1997. ISBN: 079239965X

[313] Fouskakis, D., "Stochastic optimization methods for cost-effective quality assessment in health," *PhD dissertation, Department of Mathematical sciences, University of Bath, UK*, 2001

[314] Laguna, M., Barnes, J. W., and Glover, F., "Tabu search methods for a single machine scheduling problem," *Journal of Intelligent Manufacturing*, Vol. 2, pp. 63-74, 1991

[315] Skorin-Kapov, J., "Tabu search applied to the quadratic assignment problem," *ORSA Journal on Computing*, Vol. 2, pp. 33-45, 1990

[316] Laguna, M., and Gonzalez-Velarde, J. L., "A search heuristic for just-in-time scheduling in parallel machines," *Journal of Intelligent Manufacturing*, Vol. 2, pp. 253-260, 1991

[317] Oliveira, S., and Stroud, G., "A parallel version of tabu search and the path assignment problem," *Heuristics for Combinatorial Optimization*, Vol. 4, pp. 1-24, 1989

[318] Jaumard, B., Hansen, P., and Poggi di Aragao, M., "Column generation methods for probabilistic logic," *ORSA Journal on Computing*, Vol. 3, pp. 135-148, 1991

[319] De Werra, D., and Hertz, A., "Tabu search techniques: a tutorial and an application to neural networks," *OR spectrum*, Vol. 11, pp. 131-141, 1989

[320] Malek, M., Guruswamy, M., Pandya, M., and Owens, H., "Serial and parallel simulated annealing and tabu search algorithms for the traveling salesman problem," *Annals of Operations Research*, Vol. 21, pp. 59-84, 1989

[321] Hertz, A., and de Werra, D., "sing tabu search techniques for graph coloring," *Computing*, Vol. 29, pp. 345-351, 1987

[322] Taillard, E., "Some efficient heuristic methods for the flow shop sequencing problem," *European Journal of Operational Research*, Vol. 47, pp. 65-74, 1990

[323] Widmer, M., "Job shop scheduling with tooling constraints: a tabu search approach," *Journal of Operations Research Society*, Vol. 42, pp. 75-82, 1991

[324] Bland, J. A., and Dawson, G. P., "Tabu search and design optimization," *Computer-Aided Design*, Vol. 23, pp. 195-202, 1991

[325] Beyer, D. A., and Ogier, R. G., "Tabu learning: a neural network search method for solving nonconvex optimization problems," *IEEE and INNS International Joint Conference on Neural Networks, Singapore*, Vol. 2, pp. 953-961, 1991

[326] Dorigo, M., "Optimization, Learning and Natural Algorithms," *PhD thesis, Politecnico di Milano, Italy*, 1992

[327] Colorni, A., Dorigo, M., and Maniezzo, V., "Distributed Optimization by Ant Colonies," *Actes de la première conférence européenne sur la vie artificielle, Paris, France, Elsevier Publishing*, pp. 134-142, 1991

[328] Goss, S., Aron, S, Deneubourg, J.-L., and Pasteels, J.-M., "Self-organized shortcuts in the Argentine ant," *Naturwissenschaften*, Vol. 76, pp. 579-581, 1989

[329] Deneubourg, J.-L., Aron, S., Goss, S., and Pasteels, J.-M., "The self-organizing exploratory pattern of the Argentine ant," *Journal of Insect Behavior*, Vol. 3, pp. 159, 1990

[330] Liao, W.-H., Kao, Y., and Wu, R.-T., "Ant colony optimization based sensor deployment protocol for wireless sensor networks," *Expert Systems with Applications*, Vol. 38, pp. 6599-6605, 2011

[331] Grassé, P.-P., "La reconstruction du nid et les coordinations inter-individuelles chez Belicositermes natalensis et Cubitermes sp. La théorie de la Stigmergie : Essai d'interprétation du comportement des termites constructeurs," *Insectes Sociaux*, No. 6, pp. 41-80, 1959

[332] Goss, S., Aron, S., Deneubourg, J.-L., and Pasteels, J. M., "Self-organized shortcuts in the Argentine ant," *Naturwissenschaften*, Vol. 76, pp. 579-581, 1989

[333] Marsh, L., and Onof, C., "Stigmergic epistemology, stigmergic cognition," *Cognitive Systems Research*, 2007. doi: 10.1016/j.cogsys.2007.06.009

[334] Bonabeau, E., "Editor's Introduction: Stigmergy," *Special issue of Artificial Life on Stigmergy*, Vol. 5, No. 2, pp. 95-96, 1999

[335] Deneubourg, J.-L., Aron, S., Goss, S., and Pasteels, J. M., "The self-organizing exploratory pattern of the Argentine ant," *Journal of Insect Behavior*, Vol. 3, pp. 159-168, 1990

[336] Dorigo, M., Maniezzo, V., and Colorni, A., "Positive feedback as a search strategy," *Technical Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy*, 1991

[337] Dorigo, M., Birattari, M., and Stützle, T., "Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique," *IEEE Computational Intelligence Magazine*, Vol. 1, No. 4, pp. 28-39, 2006

[338] Dorigo, M., Maniezzo, V., and Colorni, A., "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics--Part B*, Vol. 26, No. 1, pp. 29-41, 1996

[339] Yaseen, S. G., and Al-Slamy, N. M. A., "Ant Colony Optimization," International Journal of Computer Science and Network Security, Vol. 8, No. 6, pp. 351-357, 2008

[340] Rong-Long, W. Xiao-Fan, Z., Li-Qing, Z., and Ze-Wei, X., "A multi-colony ant system for combinatorial optimization problem," *International Journal of Computational Intelligence and Applications*, Vol. 11, No. 2, pp. 1-8, 2012

[341] Wong, T. N., Sicheng, Z., Gong, W., and Luping, Z., "Integrated process planning and scheduling – multi-agent system with two-stage ant colony optimization algorithm," *International Journal of Production Research*, Vol. 50, No. 21, pp. 6188-6201, 2012

[342] Ding, Q. L., Hu, X. P., and Wang, Y. Z., "An improved ant colony optimization and its application to vehicle routing problem with time windows," *Neurocomputing*, Vol. 98, pp. 101-107, 2012

[343] Liao, T. W., Kuo, R. J., and Hu, J. T. L., "Hybrid ant colony optimization algorithms for mixed discrete-continuous optimization problems," *Applied Mathematics and Computation*, Vol. 219, No. 6, pp. 3241-3252, 2012

[344] ANTS, International Conference on Swarm Intelligence, Institut de Recherches Interdisciplinaires et de Developpements en Intelligence Artificielle (Interdisciplinary and Development Research Institute on Artificial Intelligence), Belgium

[345] Swarm Intelligence Journal, Editor-in-Chief Marco Dorigo, ISSN: 1935-3812, No. 11721

[346] Dorigo, M., and Di Caro, G., "The Ant Colony Optimization meta-heuristic", *New Ideas in Optimization, Corne, D., et al., Editions., McGraw Hill, London, UK*, pp. 11-32, 1999

[347] Dorigo, M., Di Caro, G., and Gambardella, L. M., "Ant algorithm for discrete optimization", *Artificial Life*, Vol. 5, No. 2, pp. 137-172, 1999

[348] Stützle, T., and Hoos, H. H., "MAX MIN Ant System," *Future Generation Computer Systems*, Vol. 16, pp. 889-914, 2000

[349] Dorigo, M., and Gambardella, L. M., "Ant Colony System : A Cooperative Learning Approach to the Traveling Salesman Problem," *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, pp. 53-66, 1997

[350] Hu, X., Zhang, J., and Li, Y., "Orthogonal methods based ant colony search for solving continuous optimization problems," *Journal of Computer Science and Technology*, Vol. 23, No. 1, pp.2-18, 2008

[351] George, A., and Rajakumar, B. R., "Fuzzy Aided Ant Colony Optimization Algorithm to Solve Optimization Problem," *Intelligent Informatics, Advances in Intelligent Systems and Computing*, Vol. 182, pp. 207-215

[352] Di Caro, G., and Dorigo, M., "AntNet: Distributed stigmergetic control for communications networks," *Journal of Artificial Intelligence Research*, Vol. 9, pp. 317-365, 1998

[353] Aljanaby, A., Ku-Mahamud, K. R., and Norwawi, N. Md., "Interacted Multiple Ant Colonies Optimization Approach to Enhance the Performance of Ant Colony Optimization algorithms," *Computer and Information Science*, Vol. 3., No. 1, 2010

[354] Mary, L. M., and Kesavan, N. N., "Optimal design of multi-machine power system stabilizer using robust ant colony optimization technique," *Transactions of the nstitute of Measurement and Control*, Vol. 34, No. 7, pp. 829-840, 2012

[355] Tyagi, S. K., Kai, Y., and Verma, A., "Non-discrete ant colony optimization (NdACO) to optimize the development cycle time and cost in overlapped product development," *International Journal of Production Research*, Vol. 51, No. 2, pp. 346-361, 2013

[356] Mohan, B. C., and Baskaran, R., "Review: A survey: Ant Colony Optimization based recent research and implementation on several engineering domain," *Expert Systems With Applications*, Vol. 39, No. 4, pp. 4618-4627, 2011

[357] Lopez-Ibanez, M., and Stützle, T., "The Automated Design of Multiobjective Ant Colony Optimization Algorithms," *IEEE Transactions on Evolutionary Computation*, Vol. 16, No. 6, pp. 861-875, 2012

[358] Gutjahr, W. J., "A Graph-based Ant System and its convergence," *Future Generation Computer Systems*, Vol. 16, No. 8, pp. 873-888, 2000

[359] Dorigo, M., and Blum, C., "Ant colony optimization theory: A survey," *Theoretical Computer Science*, Vol. 344, No. 2-3, pp. 243-278, 2005

[360] Krishnanand, K. N., and Ghose, D., "Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions," *Swarm Intelligence*, Vol. 3, No. 2, pp. 87-124, 2009

[361] Eberhart, R. C., and Kennedy, J., "A New Optimizer Using Particle Swarm Theory," *IEEE 6th International Symposium on Micro Machine and Human Science*, 1995

[362] Wachowiak, M. P., Smolikova, R., Zheng, Y., Zurada, J. M., and Elmaghraby, A. S., "An approach to multimodal biomedical image registration utilizing particle swarm optimization," *IEEE Trans. on Evolutionary Computation*, Vol. 8, No. 3, pp. 289-301, 2004

[363] Shi, Y., and Eberhart, R. C., "Empirical study of Particle Swarm Optimization," *IEEE Congress on Evolutionary Computation*, pp. 1945-1950, 1999

[364] Davis, L., "Handbook of Genetic Algorithms," *Van Nostrand Reinhold, New York, NY*, 1991

[365] Madar, J., Abonyi, J., and Szeifert, F., "Interactive Particle Swarm Optimization," *5th International Conference on Intelligent Systems Design and Applications*, pp. 314-319, 2005

[366] Mikki, S., and Kishk, A., "Particle Swarm Optimization: A Physics-Based Approach," *Synthesis Lectures on Computational Electromagnetics*, 2008

[367] Angeline, P. J., "Using Selection to Improve Particle Swarm Optimization," IEEE *International Conference on Evolutionary Computation*, pp. 84-89, 1998

[368] Zhang, C., Shao, H., and Li, Y., "Particle Swarm Optimization for Evolving Artificial Neural Network," *IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 4, pp. 2487-2490, 2000

[369] Kennedy, J., "The particle swarm: Social adaptation of knowledge," *IEEE International Conference on Evolutionary Computation*, pp. 303-308, 1997

[370] Mendes, R., Kennedy, J., and Neves, J., "The fully informed particle swarm: Simpler, maybe better," *IEEE Transactions on Evolutionary Computation*, Vol. 8, No. 3, pp. 204-210, 2004

[371] Suganthan, P. N., "Particle swarm optimiser with neighbourhood operator," *IEEE Congress on Evolutionary Computation*, pp. 1958-1962, 1999

[372] Mohais, A., Mendes, R., Ward, C., and Postoff, C., "Neighborhood re-structuring in particle swarm optimization," in S. Zhang, and R. Jarvis Editors, *18th Australian Joint Conference on Artificial Intelligence, Springer-Verlag, Berlin*, pp. 776-785, 2005

[373] Blackwell, T. M., and Bentley, P. J., "Don´t push me collision-avoiding swarms," *IEEE Congress on Evolutionary Computation*, pp. 1691-1696, 2002

[374] Riget, J., and Vesterstroem, J., "A diversity-guided particle swarm optimizer - the arpso," *Technical Report, Department of Computer Science, University of Aarhus*, 2002

[375] Poli, R., Di Chio, C., and Langdon, W. B., "Exploring extended particle swarms: A genetic programming approach," *Conference on Genetic and Evolutionary Computation*, pp. 169-176, 2005

[376] Liu, J., Xu, W., and Sun, J., "Quantum-behaved particle swarm optimization with mutation operator," *17th IEEE International Conference on Tools with Artificial Intelligence*, pp. 237-240, 2005

[377] Iqbal M., and Montes de Oca, M. A., "An estimation of distribution particle swarm optimization algorithm," in M. Dorigo, L. M. Gambardella, M. Birattari, A. Martinoli, R. Poli, and T. Stützle editors, *"Ant Colony Optimization and Swarm Intelligence,"* 5th International Workshop, Springer-Verlag, Berlin, pp. 72-83, 2006

[378] Kennedy, J., and Eberhart, R. C., "A discrete binary version of the particle swarm algorithm," *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 4104-4108, 1997

[379] Wang, K-P., Huang, L., Zhou, C-G., and Pang, W., "Particle swarm optimization for traveling salesman problem," *IEEE International Conference on Machine Learning and Cybernetics*, pp. 1583-1585, 2003

[380] Parsopoulos, K. E., Plagianakos, V. P., Magoulas, G. D., and Vrahatis, M. N., "Improving particle swarm optimizer by function "stretching"," *Advances in Convex Analysis and Global Optimization*, pp. 445-457, 2001

[381] Parsopoulos, K. E., Plagianakos, V. P., Magoulas, G. D., and Vrahatis, M. N., "Stretching technique for obtaining global minimizers through particle swarm optimization," *Workshop on Particle Swarm Optimization*, 2001

[382] Parsopoulos, K. E., Plagianakos, V. P., Magoulas, G. D., and Vrahatis, M. N., "Objective function "stretching" to alleviate convergence to local minima," *Nonlinear Analysis, Theory, Methods and Applications*, Vol. 47, No. 5, pp. 3419-3424, 2001

[383] Parsopoulos, K. E., and Vrahatis, M. N, "Modification of the particle swarm optimizer for locating all the global minima," *International Conference on Artificial Neural Networks and Genetic Algorithms*, pp. 324-327, 2001

[384] Salerno, J., "Using the particle swarm optimization technique to train a recurrent neural model," *IEEE International Conference on Tools with Artificial Intelligence*, pp. 45-49, 1997

[385] P. Oliveira, P., Cunha, J., and Coelho, J., "Design of pid controllers using the particle swarm algorithm," *21st International Conference on Modeling, Identification, and Control*, 2002

[386] Cockshott, A. R., and Hartman, B. E., "Improving the fermentation medium for echinocandin b production part ii: Particle swarm optimization," *Process Biochemistry*, Vol. 36, pp. 661–669, 2001

[387] Abido, M. A., "Optimal power flow using particle swarm optimization," *International Journal of Electrical Power & Energy Systems*, Vol. 2, pp. 563–571, 2002

[388] Ghoshal, S. P., "Optimization of pid gains by particle swarm optimizations in fuzzy based automatic generation control," *International Journal of Electrical Power & Energy Systems*, Vol. 72, pp. 203–212, 2004

[389] Ourique, C. O., Biscaia, E. C., and Pinto, J. C., "The use of particle swarm optimization for dynamical analysis in chemical processes," *Computers and Chemical Engineering*, Vol. 26, pp. 1783–1793, 2002

[390] Secrest, B. R., "Traveling Salesman Problem for Surveillance Mission using PSO," *Air Force Institute of Technology*, 2001

[391] Krohling, R. A., Knidel, H., and Shi, Y., "Solving numerical equations of hydraulic problems using PSO," *IEEE CEC*, 2002

[392] Kadrovach, B. A., and Lamont, G., "A particle swarm model for swarm-based networked sensor systems," *ACM Symposium on Applied Computing*, pp. 918-924, 2002

[393] Omran, M., Salman, A., and Engelbrecht, A. P., "Image classification using PSO*," 4th Asia-Pacific Conference on Simulated Evolution and Learning*, pp. 307-374, 2002

[394] Coello Coello, C. A., Luna, E. H., and Aguirre, A. H., "Use of PSO to design combinatorial logic circuits," *LNCS*, No. 2606, pp. 398-409, 2003

[395] Onwubolu, G. C., and Clerc , M., "Optimal path for automated drilling operations by a new heuristic approach using particle swarm optimization," *International Journal of Production Research*, Vol. 4, pp. 473-491, 2004

[396] Onwubolu, G. C., "TRIBES application to the flowshop scheduling problem," *New Optimization Techniques in Engineering, Springer-Verlag, Heidelberg*, pp. 517-536, 2004

[397] Pareto, V., "The Mind and Society," Vol. IV, New York: Harcourt, Brace and Company, 1935

[398] Pareto, V., "Manual of Political Economy," A. M. Kelley, 1971

[399] Barr, N., "The relevance of efficiency to different theories of society," *Economics of the Welfare State, Fifth Edition, Oxford University Press*, pp. 46, 2012. ISBN 978-0-19-929781-8

[400] Sen, A., "Markets and freedom: Achievements and limitations of the market mechanism in promoting individual freedoms," *Oxford Economic Papers*, Vol. 45, No. 4, pp. 519-541, 1993

[401] Kung, H. T., Luccio, F., and Preparata, F. P., “On finding the maxima of a set of vectors.". *Journal of the ACM*, Vol. 22, No. 4, pp. 469-476, 1975. doi:10.1145/321906.321910

[402] Parke, G., Shipley, R., and Gryz, J., “Algorithms and Analyses for Maximal Vector Computation". *VLDB Journal*, Vol. 16, pp. 5-28, 2006. doi:10.1007/s00778-006-0029-7

[403] Maginot, J., Guenov, M., Fantini, P., and Padulo, M., “A Method for Assisting the Study of Pareto Solutions in Multi-objective Optimization,” *Seventh Aviation Technology, Integration and Operations Conference, Belfast, Northern Ireland*, 2007

[404] Chankong, V., and Haimes, Y. Y., “Multiobjective Decision Making Theory and Methodology,” *North-Holland, New York*, 1983

[405] Fonseca, C. M., and Fleming, P. J., “An overview of evolutionary algorithms in multi-objective optimization,” *International Journal of Evolutionary Computation*, Vol. 3, pp. 1-16, 1995

[406] Horn, J., Nafploitis, N., and Goldberg, D. E., “A niched Parto genetic algorithm for multiobjective optimization,” *Fogel Editions, Proceedings of the First IEEE conference on Evolutionary Computation, IEEE Press, Piscataway, NJ*, pp. 82-87, 1994

[407] Wilson, B., Cappelleri, d. J., Simpson, T. W., and Frecker, M. I., “Efficient Pareto frontier exploration using surrogate approximations,” *Proceedings of the Eighth AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Long Beach, CA*, 2000

[408] Shim, M.-B., Suh, M.-W., Furukawa, T., Yagawa, G., and Yoshimura, S., “Pareto-based continuous evolutionary algorithms for multiobjective optimization,” *Engineering Computations*, Vol. 19, No. 1, pp. 22-48, 2001

[409] Mueller-Gritschneder, D., Graeb, H., and Schlichtmann, U., “A successive approach to compute the bounded Pareto front of practical multiobjective optimization problems,” *SIAM Journal on Optimization*, Vol. 20, No. 2, pp. 915-934, 2009

[410] Parsopoulos, K. E., Vrahatis, M. N., “UPSO: A Unified Particle Swarm Optimization Scheme,” *Lecture Series on Computer and Computational Sciences, International Conference of Computational Methods in Sciences and Engineering*, Vol. 1, pp. 868-873, 2004

[411] Parsopoulos, K. E., and Vrahatis, M. N., “Unified Particle Swarm Optimization for Tackling Operations Research Problems,” *IEEE Swarm Intelligence Symposium*, pp. 53-59, 2005

[412] Coello Coello, C. A., Lechurga, M. S., "MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization," *Congress on Evolutionary Computation, part of the IEEE World Congress on Computational Intelligence*, pp. 1051-1056, 2002

[413] Knowles, J., and Corne, D., "The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Pareto Multiobjective Optimisation," *Congress on Evolutionary Computation*, pp. 98-105, 1999

[414] Zitzler, E., and Thiele, L., "An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach," *Computer Engineering and Communication Networks Lab, Swiss Federal Institute of Technology, Zürich, Switzerland*, 1998

[415] Zitzler, E., Laumanns, M., and Thiele, L., "SPEA2: Improving the Strength Pareto Evolutionary Algorithm," *Computer Engineering and Networks Laboratory (TIK), Zurich, Switzerland*, 2001

[416] Zitzler, E., Laumanns, M., and Thiele, L., "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization," in K.C. Giannakoglou, and others editors, *"Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems,"* pp. 95-100, 2002

[417] Srinivas, N., and Deb, K., "Multiobjective optimization using nondominated sorting in genetic algorithm," *International Journal of Evolutionary Computation*, vol. 2, pp. 221-248, 1994

[418] Grimaccia, F., Mussetta, M., Pirinoli, P., and Zich, R. E., "Genetical Swarm Optimization (GSO): a class of Population-based Algorithms for Antenna Design," *IEEE 1st International Conference on Communications and Electronics*, pp. 467-471, 2006

[419] De Castro, L. N., "Learning and Optimization Using the Clonal Selection Principle," *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 3, pp. 239-251, 2002

[420] Rechenberg, L., "Evolution Strategy," *Computational Intelligence Imitating Life*, J. M. Zurada, R. J. Marks II, and C. J. Robinson Editors, pp. 147-159, 1994

[421] Fogel, D. B., Fogel, L. J., and Atmar, J. W., "Meta-evolutionary programming," *25th Conference on Signals, Systems, and Computers*, pp. 540-545, 1991

[422] Fogel, D. B., "Evolutionary Computation: Toward a New Philosophy of Machine Intelligence," *IEEE Press*, 1995

[423] Wang, Q., Wang, C., and Gao, X. Z., "A Hybrid Algorithm based on Clonal Selection Principle and Particle Swarm Optimization," *6th International Conference on Intelligent Systems Design and Applications*, Vol. 2, pp. 975-979, 2006

[424] Juang, C-F., "A Hybrid of Genetic Algorithm and Particle Swarm Optimization for Recurrent Network Design," *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, Vol. 34, No. 2, pp. 997-1006, 2004

[425] Konak, A., Coit, D. W., and Smith, A. E., "Multi-objective optimization using genetic algorithms: A tutorial," *Reliability Engineering and System Safety*, Vol. 91, pp. 992-1007, 2006

[426] Manual for the Operation of the Joint Capabilities Integration and Development System (JCIDS), Feb 2009

[427] Department of Defense, "Critical Infrastructure Protection," 2002

[428] S&T Stakeholders Conference, "Infrastructure and Geophysical Division," May 2007

[429] EADS, Cassidian Technology

[430] Mavris, D. N. and Kirby, M. R., "Technology Identification, Evaluation, and Selection for Commercial Transport Aircraft," presented at the 58th Annual Conference Of Society of Allied Weight Engineers, May 1999

[431] Davis, P., Bankes, S., Egner, M., "Enhancing Strategic Planning with Massive Scenario Generation Theory and Experiments," *Technical Report, The RAND Corporation, Santa Monica, CA*, 2007

[432] Jimenez, H., and Mavris, D., "Conceptual design of current technology and advanced concepts for an efficient multi-mach aircraft," *SAE Transactions*, Vol. 114, Part 1, pp. 1343-1353, 2005

[433] Buonano, M.A., Mavris, D.M., "A New Method for Aircraft Concept Selection Using Multicriteria Interactive Genetic Algorithms," *43rd AIAA Aerospace sciences Meeting and Exhibit, Reno, NV*, 2004

[434] Bandte, O., and Malinchik, S. "A Broad and Narrow Approach to Interactive Evolutionary Design – An Aircraft Design Example," *Genetic and Evolutionary Computation Conference, Seattle, WA, USA,* June 2004

[435] Roth, G., and Crossley, W., "Commercial Transport Aircraft Conceptual Design using a Genetic Algorithm Based Approach," *7th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Washington, D.C.*, 1998

[436] Kossiakoff, A., and Sweet, W. N., "Systems engineering principles and practice," *Hoboken: John Wiley & Sons Inc.*, 2003

[437] "Communication from the Commission to the Council and the European Parliament", 2004

[438] CONGRESS USA PATRIOT Act of 2001, P.L. 107-56, Sec. 1016 (e), The Library of Congress

[439] Commission of the European Communities, "Green Paper, On a European Programme for Critical Infrastructure Protection", November 2005

[440] Zou, Y., and Chakrabarty, K., "Sensor deployment and target localization based on virtual forces," *Proceedings of IEEE INFOCOM Conference*, Vol. 2, pp. 1293-1303

[441] George, W. Bush, "The National Strategy for the Physical Protection of Critical Infrastructures and Key Assets", February 2003

[442] S&T Stakeholders Conference, "Critical Infrastructure Change Detection, High Impact Technology Solution (HITS) Innovation Portfolio", May 2007

[443] Congressional Research Service, "Securing America's Borders: The Role of the Military", June 2010

[444] Congressional Research Service, "Homeland Security: Unmanned Aerial Vehicles and Border Surveillance", July 2010

[445] SWARM website: Online at http://www.swarm.org/index.php/Main_Page

[446] SWARM website: Online at http://www.swarm.org/alife7/img1.htm

[447] SWARM website: Online Swarm Documentation at http://www.swarm.org/swarmdocs-2.2/set/book468.html

[448] Collier, N., Howe, T., and North, M.: Onward and Upward, "The Transition to Repast 2.0," *First Annual North American Association for Computational Social and Organizational Science Conference, Electronic Proceedings, Pittsburgh, PA, USA,* June 2003

[449] RePast website: Online at http://repast.sourceforge.net/repast_3/index.html

[450] MASON website: Online at http://cs.gmu.edu/~eclab/projects/mason/docs/

[451] MASON website: Online at http://cs.gmu.edu/~eclab/projects/mason/

[452] Netlogo website: Online at http://ccl.northwestern.edu/netlogo/

[453] NetLogo website: Online at http://ccl.northwestern.edu/netlogo/docs/

[454] The Eclipse Foundation, "About the Eclipse Foundation": Online at http://www.eclipse.org/org/

[455] Eclipse website, Eclipse wiki FAQ: Online at http://wiki.eclipse.org/FAQ_Where_did_Eclipse_come_from%3F

[456] OSGI Alliance website: Online at http://www.osgi.org

[457] Eclipse website, Eclipse Babel Project: Online at http://babel.eclipse.org/babel/

[458] The MathWorks website, Matlab news and notes: Cleve Moler, the creator of MATLAB, "The Origins of MATLAB," December 2004, Online at http://www.mathworks.com/company/newsletters/news_notes/clevescorner/dec04.html

[459] The MathWorks website: Online at http://www.mathworks.com/products/matlab/description1.html

[460] Personal communication with homeland security experts

[461] The Google Earth website: Online at http://www.google.com/earth/index.html

[462] Custom Border Protection website: Online at http://www.cbp.gov/

[463] Department of Homeland Security website: Online at http://www.dhs.gov/files/bordersecurity.shtm

[464] Kennedy, J., and Eberhart, R., "Particle Swarm Optimization," *Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia, pp. 1942-1945, 1995

[465] Pohlheim, H., "Examples of Objective Functions," *Documentation for GEATbx version 3.7 (Genetic and Evolutionary Algorithm Toolbox for use with Matlab)*, 2005

[466] Digalakis, J. G., and Margaritis, K. G., "On Benchmarking Functions for Genetic Algorithms," *International Journal on Computer Mathematics*, Vol.0, pp; 1-27, 2000

[467] Whitley, D., Mathias, K., Rana, S., and Dzubera, J., "Building Better Test Functions," *Proceedings of the Sixth International Conference on Genetic Algorithms*, pp. 239-247, 1995. ISBN: 1-55860-370-0

[468] De Jong, K., "An analysis of the behaviour of a class of genetic adaptive systems," PhD thesis, Department of Computer Science, University of Michigan, Ann Harbor, MI, 1975

[469] Schwefel, H.-P., “Numerical Optimization of Computer Models,” *Chichester, Wiley & Sons*, 1981

[470] Rastrigin, L. A., "Extremal control systems," *Theoretical Foundations of Engineering Cybernetics Series*, Moscow, Nauka, Vol. 3, 1973

[471] Griewank, A. O., "Generalized Decent for Global Optimization," *Journal of Optimization Theory and Applications*, Vol. 34, May 1981, pp. 11-39. doi: 10.1007/BF00933356

[472] Ackley, D. H., “A Connectionist Machine for Genetic Hillclimbing,” *Boston: Kluwer Academic Publishers*, 1987

[473] Michalewicz, Z., “Gentic Algorithm + Data Structures = Evolution Programs,” *Berlin, Heidelberg, New York: Springer*, 1992

[474] Engineering Statistics Handbook

[475] Mead, R. “Statistical principles for practical application,” *Cambridge University Press*, 1988

[476] Jiju, A., “Design of Experiments for Engineers and Scientists,” *Butterworth-Heinemann, Elsevier*, 2003

VITA

Alexia Payan was born and raised in a small town in the mountainous department of the Hautes-Alpes in France. From a very young age, she has been fascinated by the brightness of the Milky Way in the dark skies of the winter nights and by the majestic condensation trails left by airplanes in the summer day skies. Since then, she has always been eager to learn more about these tiny white forms whizzing through the sky during the day, and these little shiny dots sparkling in the night sky. After high-school, she decided to pursue an education in Aerospace and Aeronautics. After three years of intensive training in Mathematics, Physics and Chemistry, in “preparatory classes for national competitive exams” in Grenoble, she was admitted to the Ecole Nationale Supérieure de l’Aéronautique et de l’Espace (SUPAERO) in Toulouse in 2004. She then spent two years acquiring new knowledge about Fluid Mechanics, Flight Dynamics and Control, Aerodynamics, Structures, and Space Sciences. Her unquenchable thirst for adventure and knowledge lead her to fly to the United States in 2006 where she entered the Aerospace Systems Design laboratory at the Georgia Institute of Technology. She pursued a Master’s in Aerospace Engineering as part of a dual-degree curriculum and eventually obtained concurrently her Master degrees from Georgia Tech and from SUPAERO in 2008. She was now ready to start her PhD under the advisement of Dr. Dimitri Mavris. She received a two-years Fellowship from the French Governmental Industry Thales Group in the Air Systems division in 2009, and a one-year Fellowship from EADS North America in 2011. During these time periods, she has been working on the development of methods and tools for the design and optimization of distributed system architectures in the context of homeland security. The present paper is the culmination of her research.