DEVELOPMENT OF A HYBRID ELECTRIC VEHICLE FOR EFFECTIVE TRAVERSAL OF CONNECTED CORRIDORS

A Thesis
Presented to
The Academic Faculty

by

Nishan Guna Sakaran

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in the
Mechanical Engineering

Georgia Institute of Technology
May 2020

DEVELOPMENT OF A HYBRID ELECTRIC VEHICLE FOR EFFECTIVE TRAVERSAL OF CONNECTED CORRIDORS

Approved by:

Dr. Michael Leamy, Advisor
School of Mechanical Engineering
Georgia Institute of Technology

Dr. David Taylor
School of Electrical Engineering
Georgia Institute of Technology

Dr. Thomas Fuller
School of Chemical and Biomolecular Engineering
Georgia Institute of Technology

Date Approved: April 27th, 2022

To the Georgia Tech EcoCAR Team

ACKNOWLEDGEMENTS

I would like to thank Nicholas Hummel for his efforts at developing the algorithms described in the first half of this thesis, and for collaborating in writing the Year 4 Final Technical Report, which is incorporated in this thesis. I would also like to thank Braeden Dickson for collaborating on the controller development for Popy and making the work in the second half of this thesis possible. Next, I'd like to thank the EcoCAR leadership that I have worked closely with, both past and present, including Jessica Britt, Sterling Smith, Brianna Thornton, Noah Schaich, Christian Free, and Eamon Flaherty for their efforts in this competition that have enabled our progress and successes. I'd like to thank the EcoCAR faculty advisors, Dr Thomas Fuller, Dr Michael Leamy, and Dr David Taylor, and our GM mentor, Ed Argalas, for enabling Georgia Tech's participation in this competition and given us this fantastic learning opportunity. I'd also like to thank all the competition sponsors and organizers who have supported this team along the way. A personal thank you to the individuals close to me who have supported me through this journey: Nisha, Trevon, Conor, Arshad, Zaeem, Sean, Zephren and Julia who have been ever-present and provided welcome distraction at times. Finally, my family, who have been nothing but supportive and loving in my education journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ACC	Adaptive Cruise Control
AVTC	Advanced Vehicle Technology Competition
CAV	Connected and Automated Vehicle
DOE	Department of Energy
DSRC	Dedicated Short Range Communication
ECU	Electronic Control Unit
ESS	Energy Storage System
GDOT	Georgia Department of Transportation
GM	General Motors
HSC	Hybrid Supervisory Controller
ISG	Integrated Starter Generator
MABX	MicroAutoBox II
MAP	Intersection Map
MGU	Motor-Generator Unit
MIL	Model-In-the-Loop
OBU	On Board Unit
PCM	Propulsion Controls and Modeling
PDR	Packet Delivery Ratio
PSI	Propulsion Systems Integration
RSU	Road Side Unit
SEFA	System Element Fault Analysis
SFIA	System Functional Interface Analysis
SPaT	Signal Phase and Timing
STPA	Systems Theoretic Process Analysis
V2X	Vehicle-to-Everything
VDP	Vehicle Development Process
VIL	Vehicle-In-the-Loop
VTS	Vehicle Technical Specifications

SUMMARY

This thesis focuses on the exploration of different schemes for the autonomous traversal of a connected corridor, and the vehicle development work done in Year 4 of the EcoCAR Mobility Challenge to enable the testing of one such scheme on Ponce De Leon Avenue in Atlanta, Georgia. Georgia Tech is one of 11 teams participating in the EcoCAR Mobility Challenge, a 4-year competition that is a part of the Advanced Vehicle Technology Competition (AVTC) series. In the fourth year of this competition, teams are challenged to refine an SAE Level II Autonomous feature (Adaptive Cruise Control), implement Vehicle-To-Everything (V2X) communication between the vehicle and connected intersections, and demonstrate a robust and reliable powertrain through extensive testing.

The effectiveness of three different schemes in autonomously traversing a Connected Corridor are explored using a cost function consisting of three costs: time taken to traverse the corridor, energy used, and driver comfort. Several key parameters will be varied to observe their impact on the three schemes' effectiveness while a Simulink model and real-world testing will be used to validate the result of the schemes.

To perform real-world testing using the hybrid vehicle developed by the Georgia Tech EcoCAR team, a comprehensive vehicle development process was followed over the four years of this competition. The development and testing methodology of a supervisory controller for a hybrid vehicle is detailed in this thesis. Finally, the organization of vehicle testing activities in the final year of the EcoCAR Mobility Challenge will be described.

CHAPTER 1. INTRODUCTION

1.1 Introduction to the EcoCAR Mobility Challenge

The EcoCAR Mobility Challenge is the current installment of the AVTC series, sponsored by the US Department of Energy (DOE), General Motors (GM), and MathWorks. This is a 4-year competition among 11 university teams from across North America to convert a 2019 Chevrolet Blazer into a hybrid vehicle to achieve better fuel economy and implement autonomous features for the Mobility-as-a-Service market. After conducting market research in Year 1 of the competition, the Georgia Tech team chose Uber and Lyft drivers in the Atlanta area as the primary target market for its vehicle. Subsequently, several vehicle architectures were modeled and simulated to identify which architecture would meet the Vehicle Technical Specifications (VTS) desired by the team for its target market. Once this architecture was approved in Year 1, Year 2 and Year 3 of this competition were dedicated to the implementation and testing of this hybrid powertrain. Work done in these two years include designing all the mechanical, thermal, and electrical systems, implementing them in the vehicle along with a basic controls strategy for each, and testing them in a closed course environment. Simultaneously, components that enable connected and automated vehicle features were integrated into the vehicle in these two years, and baseline testing was completed.

At the start of Year 4 of the competition in August 2021, a hybrid powertrain was fully integrated and functional, and the beginnings of an Adaptive Cruise Control (ACC) feature had been implemented in the vehicle. In this year, the performance and reliability of the hybrid powertrain needed to be verified through a systematic testing process, while the ACC feature needed to be refined to meet competition requirements. Teams are also tasked with implementing V2X communication capability on the vehicle in this year. Specifically, teams are tasked with equipping the vehicle to autonomously traverse connected, signalized intersections that are broadcasting Signal Phase and Timing (SPaT) and Intersection Map (MAP) data.

1.2 Introduction to vehicle architecture

Georgia Tech's hybrid vehicle architecture, selected in Year 1 of the competition, is shown in Figure 1. The team implements a P0-P4 "through-the-road" parallel hybrid vehicle architecture. The stock powercube is replaced with a 2.5 L LCV engine and a 9T30 M3D transmission from GM. The Denso Integrated Starter Generator (ISG) functions as the P0 motor generator unit (MGU) and is belted on the accessory side of the engine. The battery is a GM HEV4 Energy Storage System (ESS) and the propulsion source on the rear axle is the 50 kW Magna eRAD.

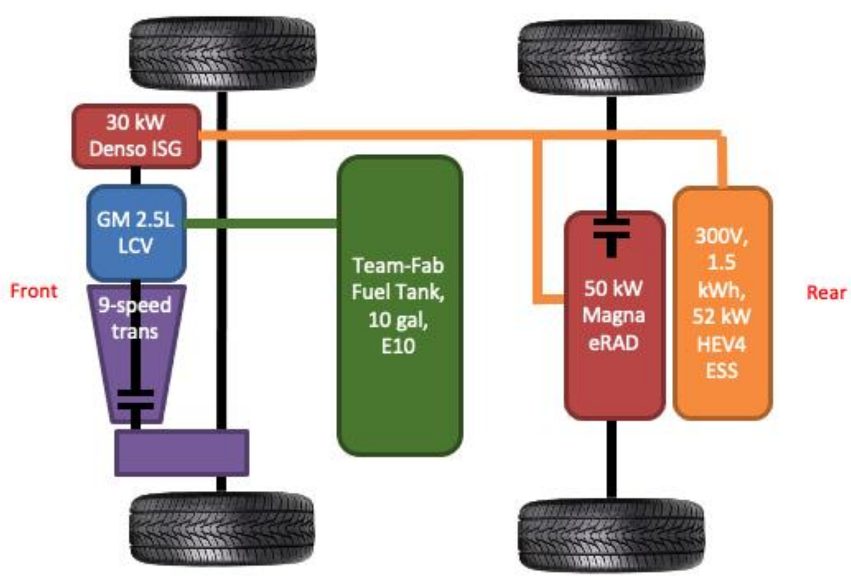

Figure 1 - Georgia Tech hybrid vehicle architecture

Georgia Tech's Connected and Automated Vehicle (CAV) architecture is shown in Figure 2. The forward perception system consists of a Bosch Mid-Range Radar (MRR) and an Intel Mobileye 6 camera. Both sensor fusion and longitudinal controller algorithms necessary to implement ACC are implemented in the primary compute unit, the Intel Tank. The Tank interfaces with actuators in the vehicle via the hybrid supervisory controller, the dSPACE MicroAutoBox II. The Cohda MK5 On-Board Unit (OBU) is a wireless radio that communicates via Dedicated Short-Range Communication (DSRC) and provides V2X capability to the team architecture. The OBU is the primary device used to receive SPaT and MAP messages from signalized intersections and communicate them to the Tank to enable autonomous traversal of signalized intersections.

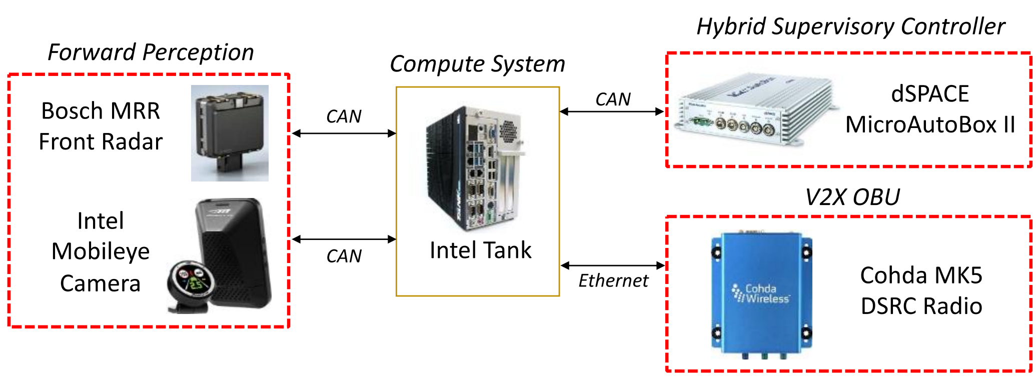

Figure 2 - Georgia Tech CAV Architecture

1.3 Ponce de Leon Avenue Connected Corridor

Ponce De Leon Avenue is a major transportation artery in Atlanta connecting the large suburban residential areas of Decatur to the populous business districts of Midtown and Downtown. According to the Georgia Department of Transportation (GDOT) website, the average daily traffic flow on this road is between 35,000 to 40,000 vehicles [1]. Figure

3 highlights 15 connected, signalized intersections along this corridor broadcasting SPaT and MAP messages. The Georgia Tech team saw it fitting to use the signal phases of intersections along this corridor as a basis for algorithm development to meet the competition V2X requirements and go above and beyond, since this would be a connected corridor frequented by the team's target market.

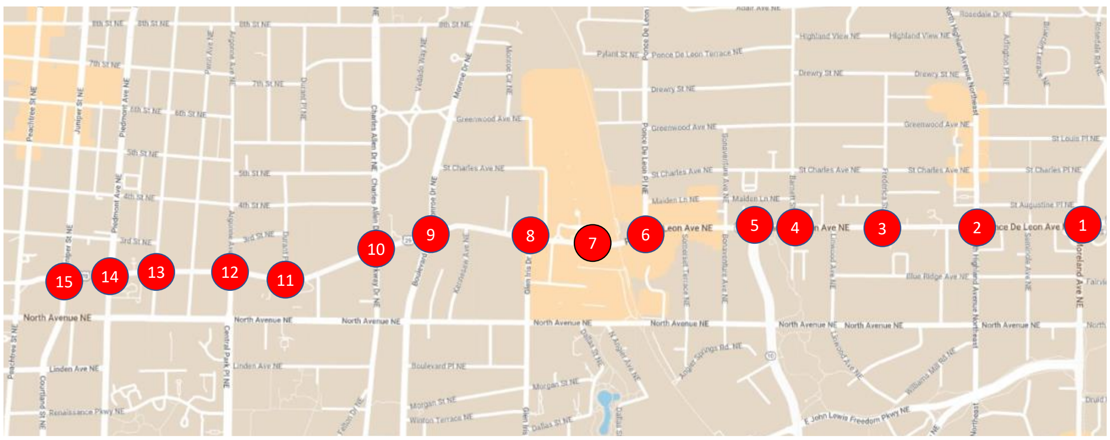

Figure 3 - Connected, signalized intersections along Ponce de Leon Avenue

To obtain representative SPaT data for each of the 15 intersections along the corridor, 15 members of the Georgia Tech EcoCAR team were stationed at the 15 intersections and recorded the SPaT simultaneously for 5 minutes, the results of which are recorded in Figure 4. This data was then fed into the models that the team used to develop and test algorithms for corridor traversal. The intersections along Ponce de Leon Avenue were used as a baseline scenario to compare different control methodologies and validate vehicle models.

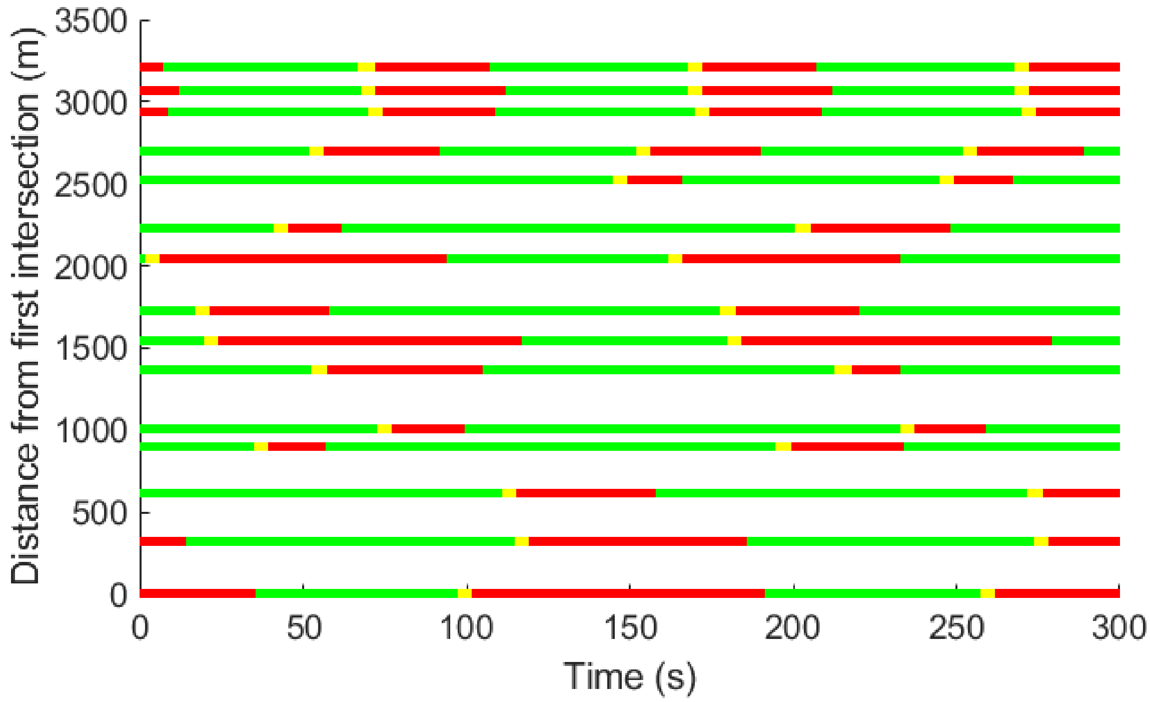

Figure 4 - SPaT along the 15 intersections of Ponce de Leon Avenue in numeric order East to West

CHAPTER 2. SCHEMES FOR AUTONOMOUS CORRIDOR TRAVERSAL

2.1 Background

The work in this chapter was submitted to the EcoCAR competition as part of the Final Technical Report (FTR) in April 2022. The FTR was a collaborative effort with Nicholas Hummel, who implemented the algorithmics described in the Proposed solutions section. The section on Model-In-The-Loop (MIL) validation, including the development of a Simulink model to validate the algorithms using a plant model, was worked on by this author, while the rest of the sections was a collaborative effort between both individuals.

2.2 Problem introduction

2.2.1.1 Problem statement

The team's goal is to develop a feature which autonomously controls the longitudinal motion of a vehicle and improves the performance of traversing a V2X connected corridor based on three driving factors: time required to traverse the corridor, energy used, and rider comfort during the traversal. Actuator limits were also considered. An assumption is made that a user-set speed limit is defined for vehicle motion. Three schemes for implementing this feature will be explored. The first is inspired by human driving behavior, referred to as the Stop or Go Scheme. The second explores optimality assuming all data can be known prior to the traversal, referred to as the Global Optimal Scheme, and the third scales down the approach used in the Global Optimal Scheme to make the solution implementable without prior knowledge of all SPaT and MAP data. It is referred to as the Local Optimal Scheme.

To evaluate each scheme for implementing this feature, different vehicle simulation models and parameters are tested. The performance of each scheme is assessed with the team collected SPaT data gathered at the Ponce de Leon connected corridor. Two simulation models are used for the vehicle: a point mass motion model, where the vehicle is abstracted to a point mass modeled by differential equations in a MATLAB simulation script, and a hybrid powertrain architecture with efficiency maps from the vehicle's actuators, in the form of a Simulink model. The point mass model is used as the initial test case for each scheme and the hybrid model is used to validate that the simplifications made in the point mass model do not significantly degrade performance. The key parameters to be explored in this work that impact the performance of the three schemes are:

1. Relative weighting of the different performance metrics
2. If any violation of the user-set speed limit is allowable, and
3. For the third scheme, the distance at which SPaT data is received for an intersection.

The point in time where the vehicle enters the corridor will also be varied to demonstrate the robustness of results to a phase shift in the SPaT cycle. Finally, a vehicle-in-the-loop test run of the Ponce de Leon connected corridor will be conducted to explore implementation hurdles.

2.2.1.2 Simplifications and constraints

To narrow the scope of this problem to something more addressable, certain simplifications and constraints were imposed. The first simplification made was modeling the road as a 1-D system, since only longitudinal motion of the vehicle is considered in this feature. The second simplification was ignoring the concepts of elevation and road grade. The last simplification made was the elimination of other vehicles in the corridor. The constraints placed on the problem exploration were limits on the vehicle's velocity range: the vehicle could not have a negative speed and the upper limit of velocity was fixed. Finally, a state limit was imposed such that any state where a vehicle is in an intersection during a red-light phase would be deemed unacceptable.

2.2.2 Proposed solutions

2.2.2.1 Stop or Go Scheme

The first scheme for implementing this feature, Stop or Go, is inspired by the decision-making process humans use when traversing an intersection. If the driver perceives that the vehicle will violate the red-light, the car is brought to a stop at the entrance to the intersection. Otherwise, the vehicle continues to travel at the desired speed limit. To perform this task in an algorithmic manner, at each time step the vehicle calculates the minimum time required to reach the next intersection. This calculation utilizes simple kinematics and a prior knowledge of vehicle acceleration and velocity limits. Once this minimum time required is known, if either of the following conditions are true, the vehicle is unable to legally traverse the intersection and must enter a stopping mode:

1. The current phase is red and the minimum time to the intersection is less than the radio reported time to the next phase.

2. The next phase is red and the minimum time to the intersection is greater than the radio reported time to the next phase.

Otherwise, the vehicle is free to continue at the user-set speed limit. This decision-making process is illustrated in Figure 5.

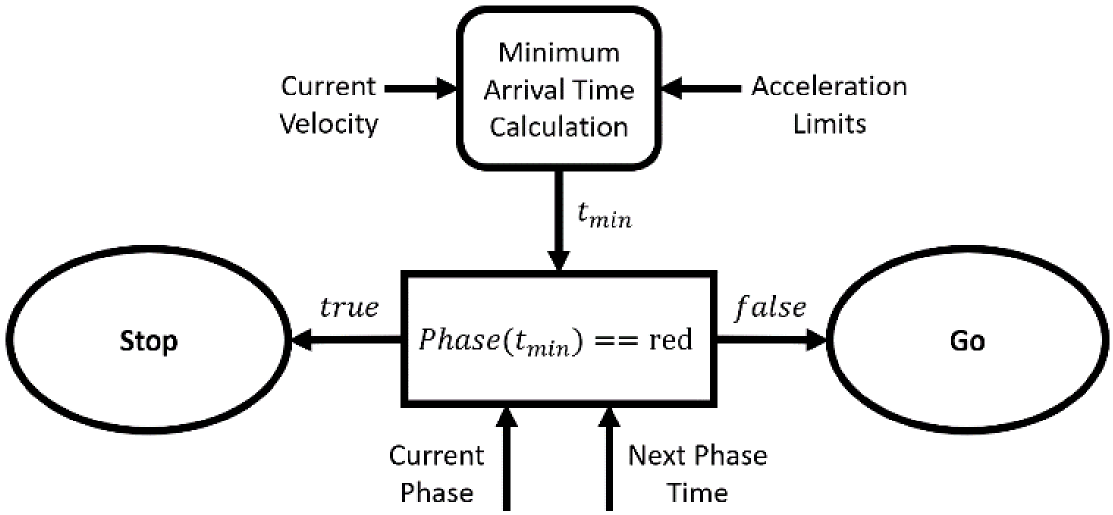

Figure 5 - Decision making process in Stop or Go Scheme

2.2.2.2 <u>Practical implementation of the Stop or Go Scheme</u>

This behavior lends itself to a two-mode architecture. The first mode, referred to as Stop Mode, handles stopping at the entrance to an intersection, and the second mode, referred to as Go Mode, regulates the vehicle's velocity to the user-set speed. Linear feedback controllers are used to implement these two modes. The Stop Mode uses a controller to regulate the distance to the entrance of an intersection, whereas the Go Mode uses a controller to regulate velocity, and these two controllers are designed and tuned independently. Both controllers in this implementation use state-space architectures augmented with integral action to allow for arbitrary pole placement and to eliminate steady-state error.

This two-mode controller requires knowledge of the vehicle velocity and distance of the vehicle to the entrance of the next intersection. The vehicle velocity can be polled from sensors at every controller time step, but the distance to the intersection is only reported by the V2X radio at a maximum rate of 10 Hz. Since the controller runs at 100 Hz and messages are sometimes missed, a state estimator is required to update the distance to the intersection based on the vehicle velocity. However, when a new SPaT message is received, the distance is updated to that value.

2.2.2.3 Global Optimal Scheme

The Stop or Go Scheme is an intuitive solution to the problem of traversing a connected corridor, but it begs the question of what the best possible solution is. If all SPaT and MAP information is known prior to the traversal of the intersection and there were infinite computational resources available, one could explore every possible velocity trajectory and measure the cost. The trajectory that yields the lowest cost is the best any traversal strategy could achieve. In the continuous time space, this is an infinite dimensional problem and is not feasible to solve. One option to address this is to analytically solve the problem. However, with all the nonlinearities and state constraints introduced in this problem, this approach quickly becomes comparably complex to an exhaustive search. Instead, the problem space is discretized and a search algorithm is implemented with intelligent search termination criteria to significantly reduce computational cost.

In this approach, the knowledge of the minimum costs to reach prior states are stored. Using this information, the search algorithm can search only trajectories that have a chance of being better than the current solution. A visualization of this discretization is shown in Figure 6. The final displacement is denoted by x_f, and the spatial step is *dx.* The final time is denoted by t_f, and the spatial step is *dt.*

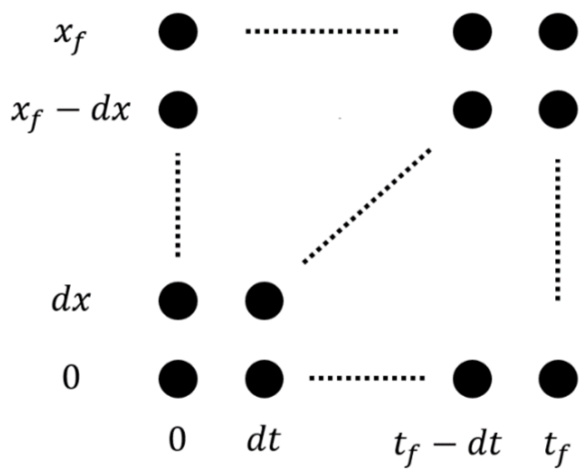

Figure 6 - Visualization of discretization of search algorithm problem space

The time step *dt* and the distance step *dx* dictate the resolution and size of the discrete space. Two meshes of this same size are stored as part of the algorithm. The first is the light matrix, which holds the phase of the light at a given position and time. If there is no light at a location in the light matrix, a 0 is stored. The second is the cost matrix, which holds the minimum cost found to reach a given state up to that point in the simulation. The cost of each state is initialized to -1, so the search algorithm can judge if a state has been visited previously. The search algorithm then starts exploring trajectories from the initial state, storing the best cost as it goes. If a trajectory search reaches a state that has already been visited, and does so in a more expensive way, the search is terminated.

An example of one step of a trajectory search is show in Figure 7. If the vehicle was traveling at 2 m/s, with a time step of 1 s and acceleration limits of ±2 m/s^2, with a quantization of 1 m/s^2 there are 5 possible next states. The range of velocities to reach each next state is saturated each time step to stay within the velocity constraints established.

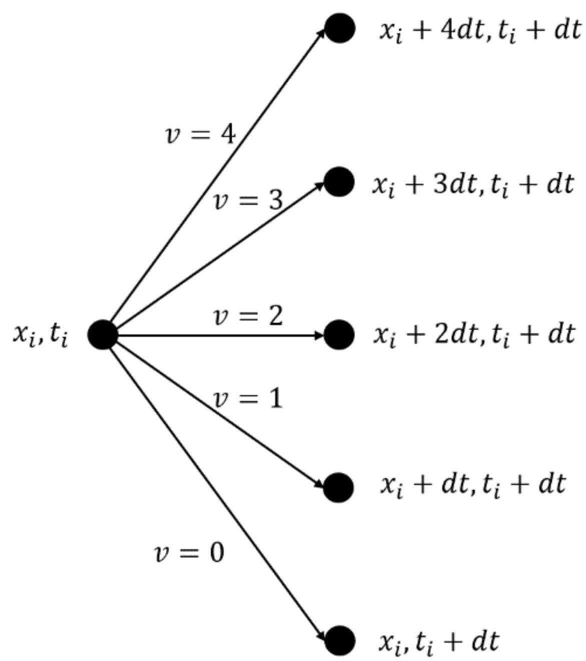

Figure 7 - Visualization of trajectory exploration from an initial state

The search algorithm looks at all possible next states given the current state and calculates a cost for each. The order in which these next states are explored is of significance to the performance of the algorithm. Initially a breadth-first search was implemented where each state parsed by the search algorithm would add all its next states to the end of a queue of next states. This led to so many states being stored simultaneously that the memory limits of the machine were exceeded. The solution was to implement a depth-first search, where an entire velocity trajectory would be explored, then the search would return to the last branching point and start a new chain. This led to a search queue length proportional to the time span of the scenario. Additionally, the order of the depth-first search proved to have significant impact. The final implementation initiates searches from the maximum possible acceleration to the minimum at each state. This leads to faster solving for certain cost functions.

The resolution of vehicle velocity states *dv* is the ratio of *dx* to *dt*. The resolution of control actions *da* is the ratio of *dv* to *dt* and dictates the number of options for the next state available to the search algorithm at the current state. As *da* becomes smaller the number of possible trajectories increases and the accuracy of the result also increases. However, this leads to huge increases in computational and memory resources. To find the feasibility of implementing this search algorithm, the runtime had to be characterized. Table 1 shows the change in both computational performance and cost for a fixed time step of 1s, and velocity step sizes of 1 and 0.1 m/s.

Table 1 - Performance of Global Optimal Scheme with different resolutions

Velocity Step (m/s)	Run Time (s)	Normalized Cost	% Error
0.1	29822.71	8.987	-
1	179.27	9.145	1.75

For only a small decrease in performance, 1.75%, the search algorithm can run over 166x faster. While this number will exhibit variance based on scenario and initial conditions, it is indicative of the relative performance loss to time gain relationship. Therefore, within the limits of the quantization, the Global Optimal Scheme can develop a near optimal velocity profile.

2.2.2.4 Practical implementation of the Global Optimal Scheme

To execute this velocity profile generated from the inverse dynamics utilized in the search algorithm in the vehicle, a controller is needed to regulate the longitudinal motion of the vehicle. The same linear feedback controller used in the Stop or Go strategy will serve as a baseline. It is desired that the vehicle follows the trajectory generated by the search algorithm as closely as possible. After some preliminary testing, it was discovered that the linear feedback state-space integral controller could not keep up with sharp acceleration requests. To address this, a feedforward component was added utilizing plant inversion in addition to the feedback control. The feedforward component can track the velocity trajectory output of the search algorithm exactly since the search algorithm already accounts for actuator limits. However, to improve robustness of the controller, the proportional and integral feedback terms on the vehicle's velocity are still used. To compare each controller, the discretized output of the search algorithm was fed as the reference command to each controller. Figure 8 shows the response of each controller to this reference command. The response of the feedback + feedforward controller is also shown with the addition of noise in the velocity signal to demonstrate robustness.

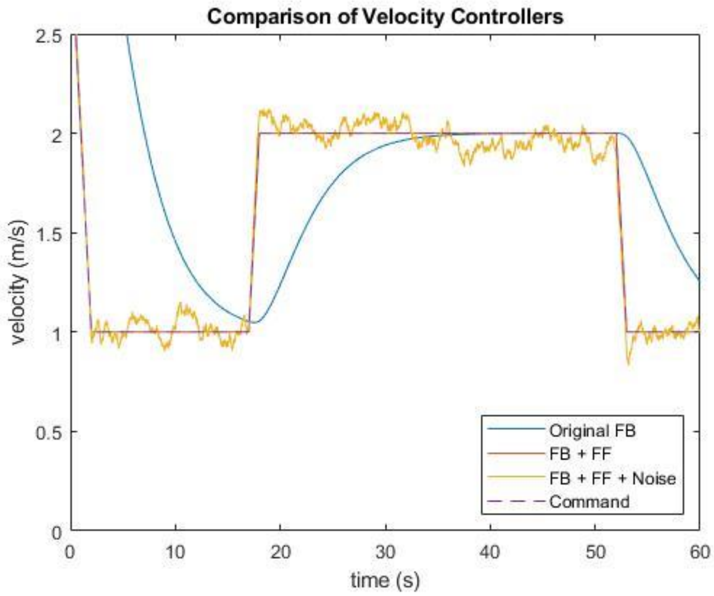

Figure 8 - Comparison of velocity controllers

The feedback + feedforward controller can be seen to match the reference command and effectively mitigate noise in the velocity signal.

2.2.2.5 Local Optimal Scheme

The Global Optimal Scheme proved to be effective at generating and tracking a velocity profile. However, it relied on knowledge of all SPaT and MAP data prior to execution. In the real world this is not implementable with the team's current architecture, since the MK5 radio only starts receiving SPaT and MAP data within 500m of an intersection. To address this, a modification was made to only execute the search algorithm for a fixed distance in front of the vehicle, which would correspond to the range at which the V2X radio receives SPaT data. The search algorithm is then called for a limited time and distance horizon at a periodic rate (e.g., once every 10 s).

2.2.2.6 Practical implementation of the Local Optimal Scheme

The same velocity regulator used in the Global Optimal Scheme is utilized to control the vehicle's velocity to the output of the search algorithm. The process of calling the search algorithm and regulating the vehicle's longitudinal motions is illustrated in Figure 9. The desired velocity is denoted by $v^*(t)$, the feedback velocity signal is denoted by $\hat{v}(t)$, and the requested torque from the vehicle is denoted by $u(t)$. The modulus function is employed to execute the search algorithm in a periodic manner.

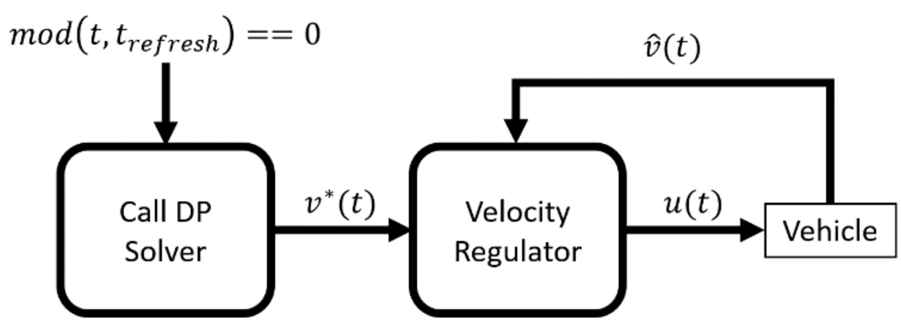

Figure 9 - Local Optimal Scheme Practical Implementation Structure

In the current implementation, the velocity trajectory generated by the periodic call of the search algorithm is fed to the velocity regulator in an open loop manner. To improve robustness in future work, feedback could be used with the search algorithm.

2.2.3 Simulation models

2.2.3.1 Point mass model

The point mass model is a first order system where the motion of the mass is affected by a control input force and parasitic drag forces. Due to its simplicity, the point mass model is used as a starting point to simulate each scheme of implementing an autonomous connected corridor traversal feature. It can be run quickly and has few parameters to tune, making it a perfect choice for initial investigations.

2.2.3.2 Simulink hybrid vehicle model

A hybrid vehicle model was developed to validate the results obtained through the point mass motion model. This model, shown in Figure 10, contains 4 main components: the signal configuration block that produces Ponce de Leon SPaT data, the CAV Control block that implements the various traversal schemes and produces a torque request which is fed to the PCM Control block that contains the hybrid controls strategy. Finally, the Passenger Car which is a plant model that models the dynamics of the 2019 Chevrolet Blazer and team-added components. This plant model uses a 1-Degree Of Freedom vehicle model using coast-down testing coefficients from EPA testing that models the dynamic powertrain load with minimal parametrization.

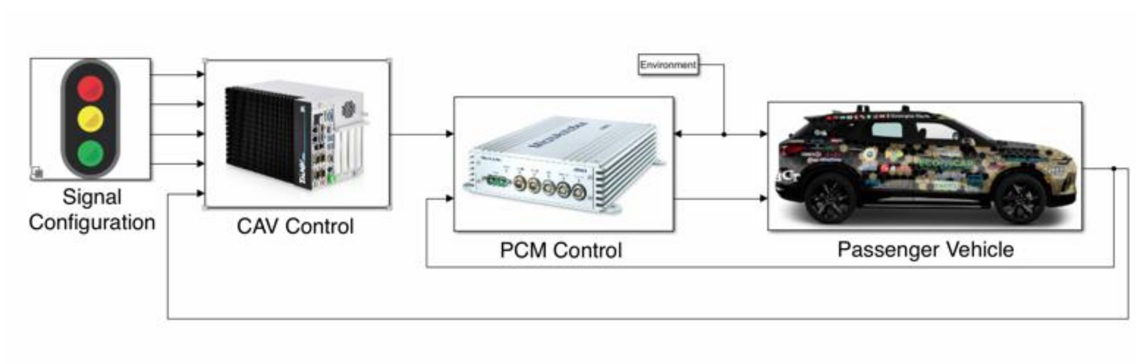

Figure 10 - Simulink hybrid vehicle model

2.2.4 Key parameters

To lend structure to the concept of optimality, a cost function was developed to account for the three driving factors discussed: time required to traverse the corridor, energy used, and rider comfort during the traversal. The cost components will hereon be referred to as the time cost, energy cost, and comfort cost respectively.

The time cost imposes a constant running cost on the time taken to traverse the corridor, where the total penalty on a traversal is the time taken to reach the final state. The energy cost imposes a cost proportional to the power expended at each time step. The energy cost includes a term for the power needed to accelerate the vehicle to the next time step's velocity as well as a term to account for the power needed to combat losses from drag forces. The comfort cost also comprises two components. The first term imposes a cost proportional to the square of the magnitude of acceleration that exceeds 1 m/s^2. The acceleration threshold for rider discomfort has been explored in several studies and lies around 1 m/s^2 [2] [3]. The second term imposes a penalty on the error between the user-set speed and the vehicle speed. This term creates a response that develops end user trust in the system, since deviations from the user-set speed are penalized.

To simplify the cost function from three components to one scalar value, weighting coefficients were assigned as follows:

- α_1 – Weight of time cost
- α_2 – Weight of energy cost
- α_3 – Weight of rider comfort cost

These three weights were modeled such that they always summed to 1, effectively creating a convex set with 2 degrees of freedom, thereby simplifying the complexity of the possible search space. The sum of the three costs multiplied by their respective weights is taken as the total cost. By increasing one cost's weight, the relative impact of the other costs was diminished. Additionally, each cost function was multiplied by a normalizing weight such that the range of each signal would not have an impact on the relative importance of its cost. For example, the vehicle's acceleration may range from 0 to 5 m/s^2, but the velocity may vary from 0 to 15 m/s. Therefore, a cost function based solely on velocity magnitude would have a normalizing weight 1/3 that of a cost function based solely on acceleration magnitude.

The expected impact of setting α_1 to 1, thereby only considering the time cost and negating the impact of the energy cost and the comfort cost, is creating a more aggressive response that rushes to the end of the corridor at the earliest possible time with no regard for energy used or any other parameter. The expected impact of setting α_2 to 1 is improved energy efficiency, which leads to a later arrival time, with less aggressive acceleration, and a lower average speed. Finally, the expected result of setting α_3 to 1 is a response with smooth acceleration, but which tries to stay at the user-set speed limit.

The allowable user-set speed limit violation is tested for 2 values, 0 mph greater than the user-set limit, and 5 mph over the user-set limit. The expected result of increasing the speed limit is an earlier arrival time and higher average speed since the controller will be incentivized to travel more quickly.

Finally, the distance at which SPaT data is received will be varied for the third scheme of implementing this feature. This decision is justified by an analysis of the V2X technology in use. The Federal Communications Commission defines 4 classes of DSRC devices, each with a maximum output power that defines its communication range as seen in Table 2.

Table 2 – Federal Communications Commission DSRC Device Classification [4]

Device Class	Max. Output Power (dBm)	Communication Zone (meters)
A	0	15
B	10	100
C	20	400
D	28.8	1000

Both the Georgia Tech vehicle and the Road-side Units (RSUs) along Ponce de Leon Avenue utilize the Cohda MK5 OBU [5], which is a Class C device [6]. However, while Class C devices are documented as having a maximum output power of 20 dBm and a maximum communication zone of 400 m, the MK5 OBU has a maximum output power of 22 dBm. This is consistent with the team's experience of detecting these intersections more than 500 m away.

The distance at which the V2X device can reliably communicate with connected intersections impacts the distance over which traversal can be optimized. Studies have shown that reliability deteriorates at longer distances. One study using Class C DSRC devices suggests that at 400 m in real world conditions, the packet delivery ratio (PDR) drops to 58% [7].

Based on this information, a maximum range of 500 m, which is representative of real-world testing, is chosen, and the impact of shorter look ahead distances is also evaluated. The expected impact of a shorter distance is lower efficiency since the controller has less knowledge about future states.

2.3 Analysis of findings

2.3.1.1 Stop or Go Scheme baseline performance

The Stop or Go algorithm will serve as a baseline comparison to the Global and Local Optimal Schemes, since it is more representative of human driving behavior and has no dependance on a cost function. This algorithm was already implemented in the vehicle for limited closed course testing, so it will also serve as a first step towards the development of the connected corridor traversal feature in the consumer vehicle. The results of the Stop or Go Scheme navigating the team collected SPaT data from the Ponce de Leon corridor is shown in Figure 11 and Table 3.

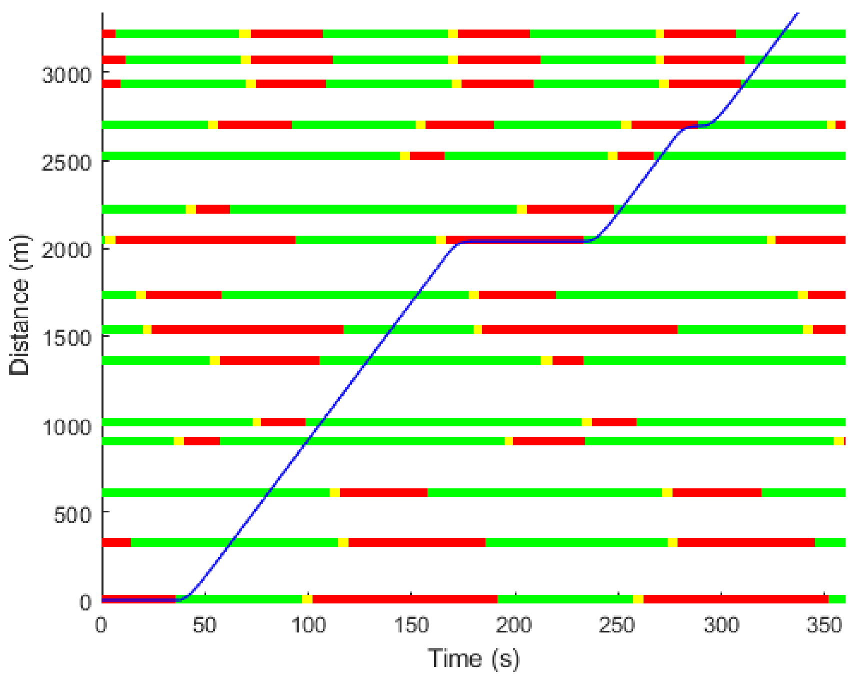

Figure 11 - Stop or Go Scheme traversal with 35 mph speed limit

Table 3 - Performance of Stop or Go Scheme with 35 mph speed limit

Traversal Time (s)	Energy Used (kJ)	Average Acceleration (m/s^2)
334.4	1227	0.0468

This scheme behaves similarly to a regular human driver where the user-set speed is followed unless it is determined that a red light would be violated. In this case the vehicle is slowed to a stop at the intersection until a green phase is received. With this baseline established, the parameters discussed previously are varied to identify their impact on traversal effectiveness as measured by time, energy, and average acceleration.

2.3.2 Effect of varying cost function weights

2.3.2.1 Global Optimal Scheme

The Global Optimal Scheme can find within the approximation of the state discretization the best possible velocity profile of the vehicle for a given cost function and knowledge of all SPaT information *a priori.* To visualize the impact of each of the three cost functions the Ponce de Leon corridor was tested with the Global Optimal Scheme for each weight, α_1, α_2, α_3 set to 1, with the others set to 0. In addition, the results where each cost had equal weight is evaluated. The results are shown in Figure 12 while the performance metrics are summarized in Table 4.

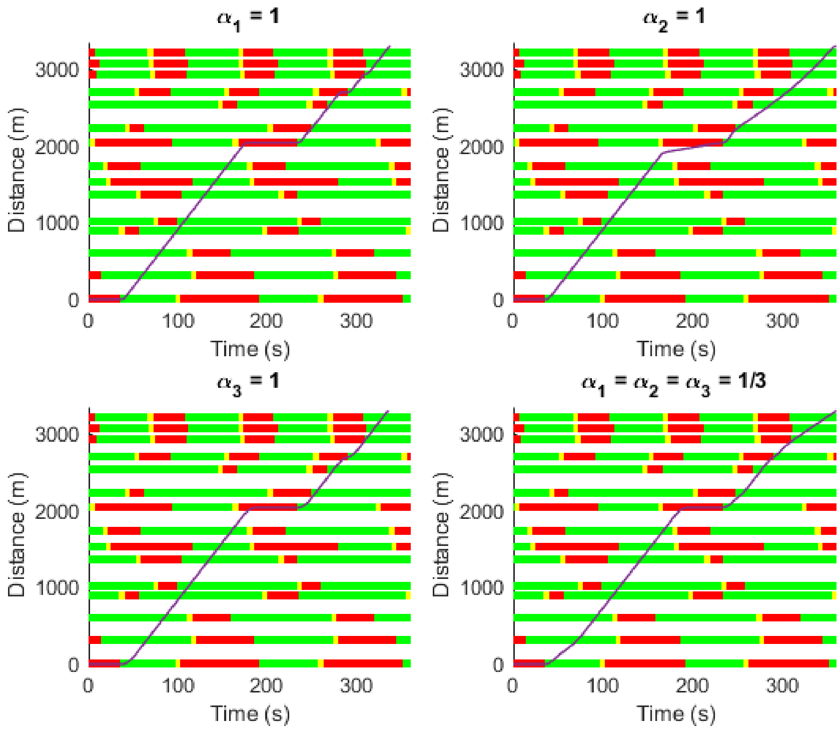

Figure 12 - Global Optimal Scheme traversals with 35 mph speed limit and varying α

Table 4 - Performance of Global Optimal Scheme with 35 mph speed limit and varying α

α_1	α_2	α_3	Traversal Time (s)	Energy Used (kJ)	Average Acceleration (m/s^2)
1	0	0	333.0	1246	0.0470
0	1	0	360.0	817	0.0028
0	0	1	334.0	1156	0.0449
1/3	1/3	1/3	359.9	831	0.0116

When time, α_1, is the only cost, the response of the controller is to rush to each light as fast as possible. In terms of the search algorithm, the cost to reach any state is just the time of that state. Therefore, any path that ends at the fastest path from the last red light to the target distance would have the optimal cost. Due to the nature of the search order and the implementation of the algorithm, this returns a very aggressive response.

When energy, α_2, is the only cost, the controller's response is to get to the goal at the last time allowed. This leads to less energy lost in actuator effort and parasitic losses. It is interesting to note that the controller favors cruising at a relatively high speed after it leaves the first light. This is due to the impact of actuator effort on the energy cost. When velocity is constant, the vehicle only must expend energy to counter parasitic drag forces.

When comfort, α_3, is the only consideration, the controller gently accelerates to a speed near the limit and travels at that speed for the rest of the corridor. The impact of the quadratic cost on acceleration magnitude can be clearly seen when the vehicle departs and brakes for lights; the curves are much gentler, since the controller is incentivized to always use the minimal allowed acceleration. Additionally, the cost term forcing the velocity toward the user-set speed has a clear impact on the cruising speed of the vehicle. With no drag forces considered, the vehicle incurs no penalty for cruising constantly at the user-set speed. The result is a higher average acceleration than the case where only energy is considered, but a much earlier arrival time.

When all costs are combined with equal weights, we see a blend of the responses from each in isolation. The smoothing impact of the energy cost and acceleration component of the comfort cost can be seen moderating the aggressive response of the time cost. Additionally, the user-set speed violation cost can be seen in the segment from roughly 75 to 175 s, where the vehicle is free to cruise without incurring that cost and without impacting the time it will clear the next light at approximately 2000 m.

2.3.2.2 Local Optimal Scheme

Using a maximum speed of 35 mph, and a distance horizon of 500 m, the cost function weights are varied to evaluate its impact on the Local Optimal Scheme's performance. The results are shown in Figure 13 and Table 5.

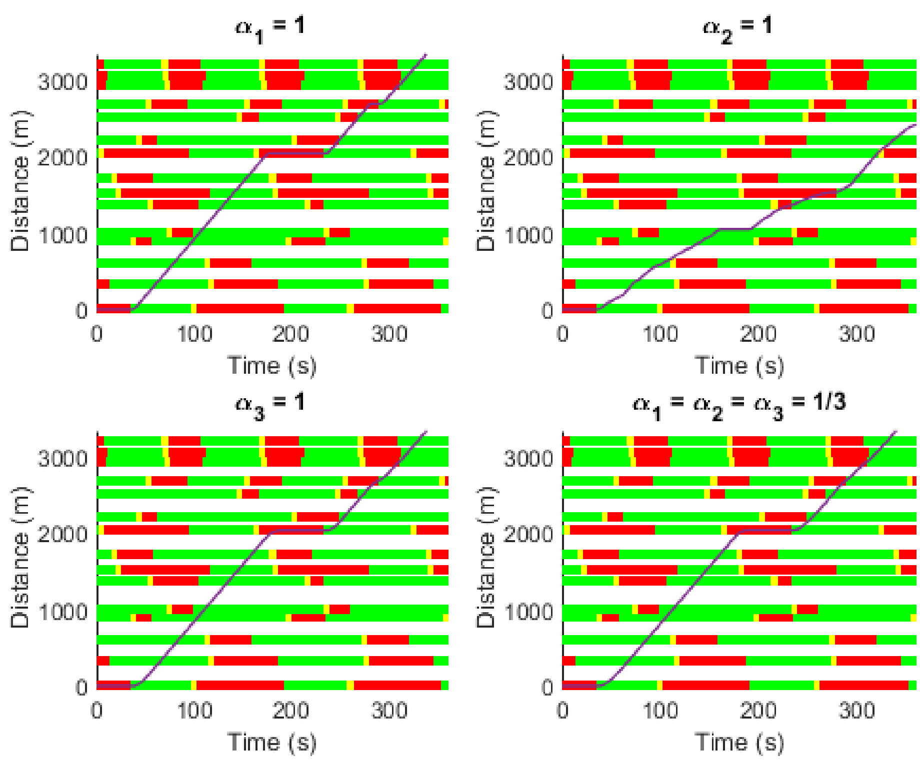

Figure 13 - Local Optimal Scheme traversals with 35 mph speed limit, 500 m distance horizon and varying α

Table 5 - Performance of Local Optimal Scheme with 35 mph speed limit, 500 m distance horizon and varying $\boldsymbol{\alpha}$

α_1	α_2	α_3	Traversal Time (s)	Energy Used (kJ)	Average Acceleration (m/s^2)
1	0	0	334.0	1267	0.0449
0	1	0	360.0*	596*	0.0139
0	0	1	334.2	1153	0.0449
1/3	1/3	1/3	336.2	1138	0.0446

*Did not make it to end distance goal

When time, α_1, is the only cost in the Local Optimal Scheme, the response of the controller is like the Global Optimal Scheme, to rush to each light as fast as possible, returning a very aggressive response that uses the most energy and has the highest average acceleration of the four scenarios.

When energy, α_2, is the only cost, the algorithm's response is to moderate acceleration heavily, leading to persistent violation of the user-set speed, including a period being completely stationary at 1000 m, presumably to avoid stopping at an intersection at 1400 m. The impact of the limited distance horizon is evident here since the response does not even reach the target distance within the simulation time.

When comfort, α_3, is the only consideration, the controller behaves similarly to the first case. The impact of the quadratic cost on acceleration magnitude can be clearly seen once again when the vehicle departs and brakes for lights, resulting in smoother acceleration compared to the first case.

When equal weights are used for all costs, the response largely matches the $\alpha_1 = 1$ and $\alpha_3 = 1$ cases. This is because the improvement to energy usage that is achieved in the $\alpha_2 = 1$ case is far outweighed by the resulting cost of violating the user-set speed and the traversal time cost. However, a glimpse of the impact of the α_2 is seen in the lower average acceleration compared to the third case – the acceleration is moderated even more than before to reduce energy usage. The overall form of the response is like the Global Optimal Scheme, and the characteristic smoothing is still seen in the approaches and exits from intersections.

2.3.3 Effect of allowing speed limit violations

In many of the trajectories, the vehicle arrives at an intersection soon after it turns red when it might have succeeded in traversing the intersection without stopping if it was able to increase its speed marginally. The impact on performance metrics of introducing a maximum speed of 40 mph, 5 mph greater than the user-set speed of 35 mph, is explored in this section.

2.3.3.1 Stop or Go Scheme

Since the Stop or Go Scheme does not differentiate between the user-set speed and the speed limit, it can serve as a baseline representative of a human's behavior if told the speed limit of a road was 40 mph. Figure 14 and Table 6 illustrate the performance of the Stop or Go Scheme when a maximum speed of 40 mph is introduced. The traversal time using the Stop or Go Scheme improves by 30% compared to the 35-mph case by eliminating the need to stop for an extended period at a red light. However, the energy used increases by 20% and the average acceleration increases by 64%. These increases are driven by the higher average speed and the presence of an intersection that causes the vehicle to slow significantly.

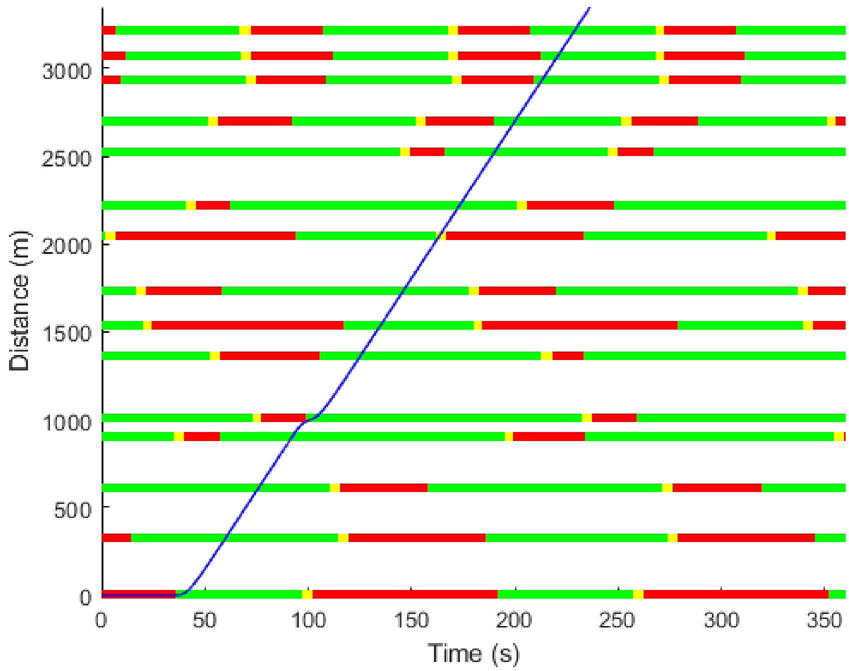

Figure 14 - Stop or Go Scheme traversal with 40 mph speed limit

Table 6 - Performance of Stop or Go Scheme with 40 mph speed limit

Traversal Time (s)	Energy Used (kJ)	Average Acceleration (m/s^2)
233.6	1465	0.0766

2.3.3.2 Global Optimal Scheme

Figure 15 and Table 7 illustrate the performance of the Global Optimal Scheme when a 40 mph speed limit is introduced for varying α. Compared to the 35 mph speed limit case, the traversal time is faster in every case, but the energy used is higher in every case as well due to the higher average speed that results in greater losses to drag forces.

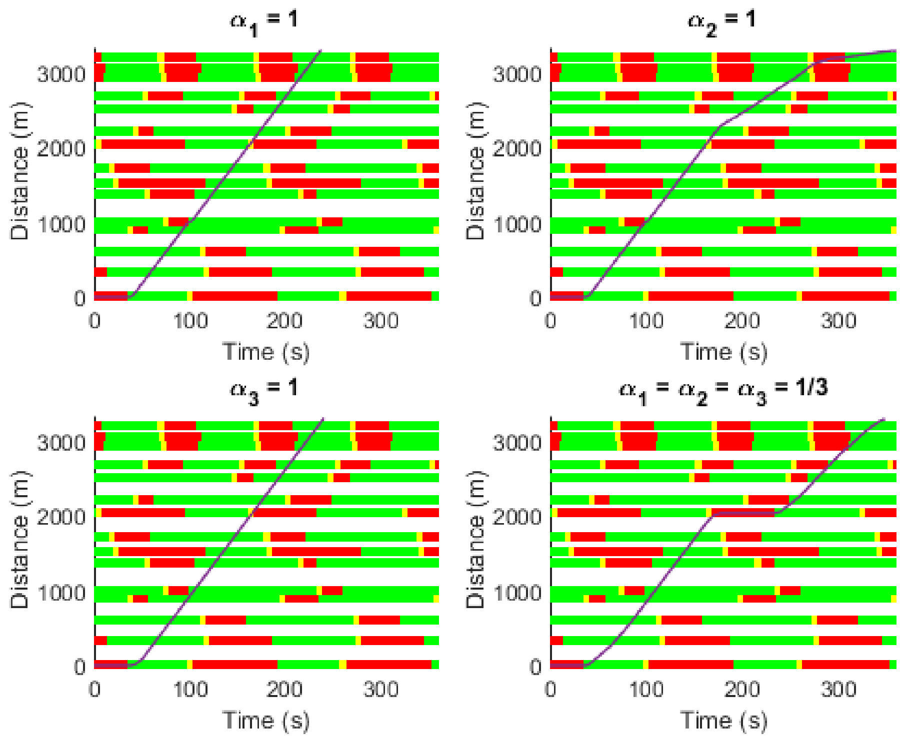

Figure 15 - Global Optimal Scheme traversals with 40 mph max speed and varying α

Table 7 - Performance of Global Optimal Scheme with 40 mph speed limit and varying α

α_1	α_2	α_3	Traversal Time (s)	Energy Used (kJ)	Average Acceleration (m/s^2)
1	0	0	235.8	1396	0.0721
0	1	0	360.0	892	0.0028
0	0	1	249	1270	0.0602
1/3	1/3	1/3	344.6	933	0.0156

When time, α_1, is the only cost in the Global Optimal Scheme, the traversal time is expected to be at a minimum. However, the Stop or Go scheme outperforms the Global Optimal Scheme, traversing the intersection 2.2 seconds faster. This discrepancy is predominantly due to the discretization of velocities in the Global Optimal Scheme, where the 40 mph (17.88 m/s) speed limit is truncated to 17 m/s (38 mph), giving the stop and go scheme a 2 mph speed advantage for the duration of the relatively constant-speed traversal. To address this discrepancy, the resolution of the discrete space used in the search algorithm could be increased at the cost of computation time.

When energy, α_2, is the only cost, a traversal with the same energy used as the 35 mph speed limit case, or better, is expected, since there is no penalty associated with deviations from the speed limit. However, the energy used in this traversal is 9% greater than that in the 35-mph speed limit traversal. Similarly, when comfort, α_3, is the only cost, the performance in the average acceleration metric is expected to be the same or better than the 35 mph speed limit traversal. However, the average acceleration is 34% higher in the 40 mph speed limit traversal.

When all costs are combined with equal weights, the traversal time improves by 4% over the 35 mph speed limit traversal, but the energy used worsens by 15% and the average acceleration increases by 34%.

These results are unexpected and suggest that even in the absence of the time cost, the scheme exhibits an affinity for a quicker traversal when a higher maximum speed is possible, even when this result is not optimal. This result is believed to be due to the current searching method implemented in the search algorithm and will be investigated as part of future work.

2.3.3.3 Local Optimal Scheme

Figure 16 and Table 8 show the response of the Local Optimal Scheme for varying values of α_1, α_2, and α_3 when the speed limit is raised by 5 mph to 40 mph. The general shape of the response closely follows the behavior of the Global Optimal Scheme when only time or rider comfort are considered. A large deviation from the Global Optimal Scheme appears when only the energy is considered. Just as in the case when the speed limit was 35 mph shown in Figure 13, the response only considers energy cost over a finite horizon and does not consider a final time. Due to these factors the response does not make it all the way through the corridor. This shows that the Local Optimal Scheme has higher sensitivity to the cost function weights than the speed limit relative to the Global Optimal Scheme.

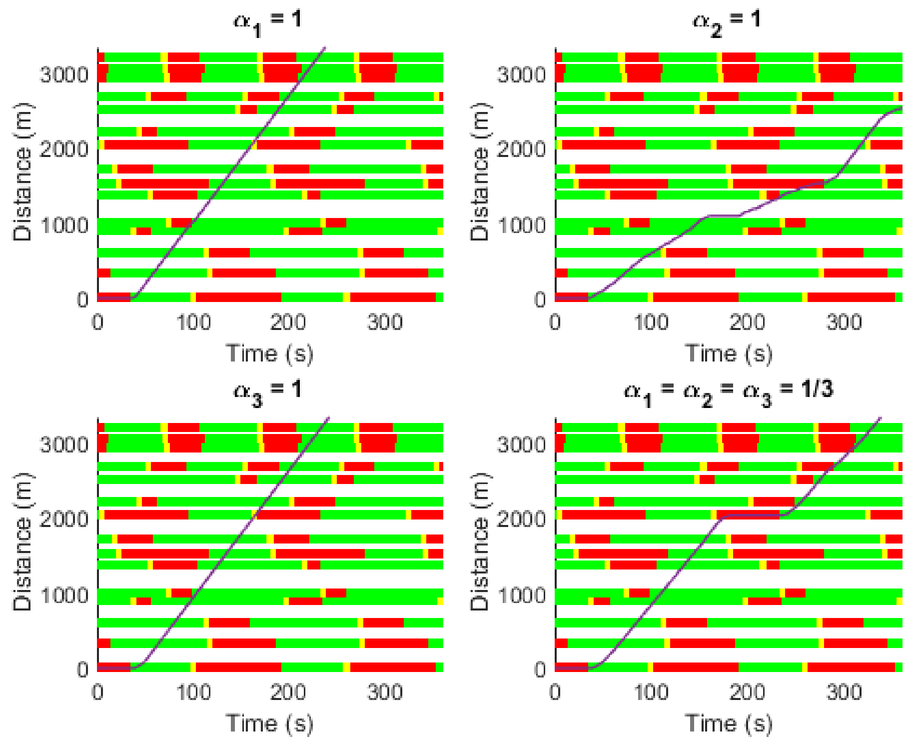

Figure 16 - Local Optimal Scheme traversals with 40 mph max speed and varying α

Table 8 - Performance of Local Optimal Scheme with 40 mph speed limit and varying α

α_1	α_2	α_3	**Traversal Time (s)**	**Energy Used (kJ)**	**Average Acceleration (m/s^2)**
1	0	0	234.7	1399	0.0724
0	1	0	360.0*	583*	0.0083*
0	0	1	238.6	1392	0.0712
1/3	1/3	1/3	334.6	1205	0.0477

*Did not make it to end distance goal

2.3.4 *Effect of varying distance horizon*

Figure 17 and Table 9 illustrate the sensitivity of the Local Optimal Solver to the distance at which it receives SPaT data. This distance is varied from 500 m to 100 m, which are representative bounds of the maximum and minimum distances where the MK5 radio first received intersection data when driving in Atlanta. All metrics: time, energy, and comfort, can be seen to improve as the time horizon distance increases. This result agrees with the intuition that the more data the search algorithm under the hood of this scheme is armed with, the better it can do. However, while these results approach the performance of the Global Optimal Scheme, it still falls short of matching it. It does outperform the Stop or Go Scheme results shown in Table 3 in energy used for all tested values of the distance horizon and in rider comfort for distance horizons between 100 and 400 m.

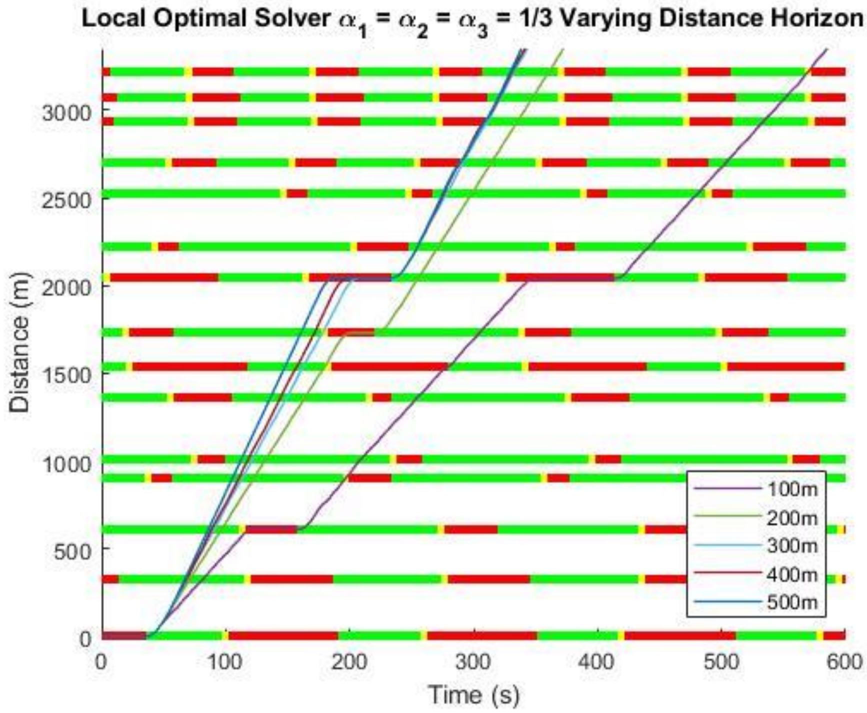

Figure 17 - Local Optimal Scheme traversals with 35 mph max speed and varying distance horizons

Table 9 - Performance of Local Optimal Scheme traversals with 35 mph max speed and varying distance horizons

Distance Horizon (m)	Traversal Time (s)	Energy Used (kJ)	Average Acceleration (m/s^2)
100	579.0	560.9	0.0121
200	368.0	838.7	0.0299
300	339.2	959.8	0.0348
400	337.4	999.5	0.0352
500	335.2	1142.2	0.0448

2.3.5 Robustness of schemes to different SPaT sequences

To evaluate the robustness of the different schemes to different SPaT sequences, each scheme was run with start times varying from 0 s to 100 s in 20 s increments as seen in Figure 18, Figure 19, and Figure 20, . The phase of the first intersection is manipulated to vary the start times, where the phase is indicated as red until the desired start time. At the start time, a green phase is indicated for the scheme to begin the traversal. The mean and standard deviation of the three-performance metrics for the six traversals for each scheme are summarized in Table 10.

Table 10 - Mean and standard deviation of performance metrics across six starting times for each scheme

Scheme	Traversal Time (s)		Energy Used (kJ)		Average Acceleration (m/s^2)	
	μ	σ	μ	σ	μ	σ
Stop or Go	368	103	1214	24	0.045	0.013
Global	417	84	762	83	0.006	0.003
Local	387	85	1127	36	0.040	0.007

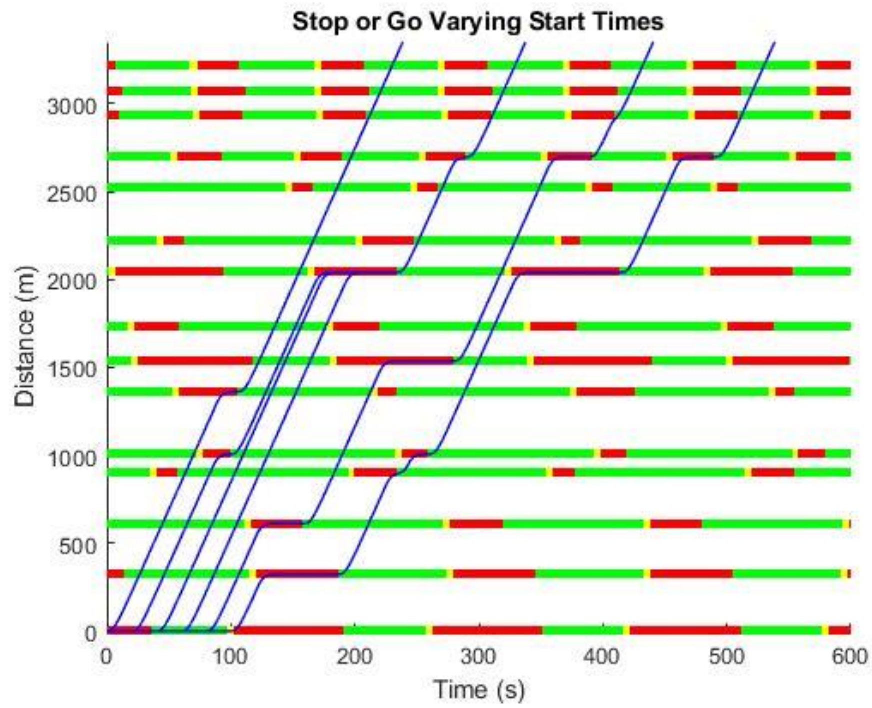

Figure 18 - Traversals with six varying start times for the Stop or Go Scheme

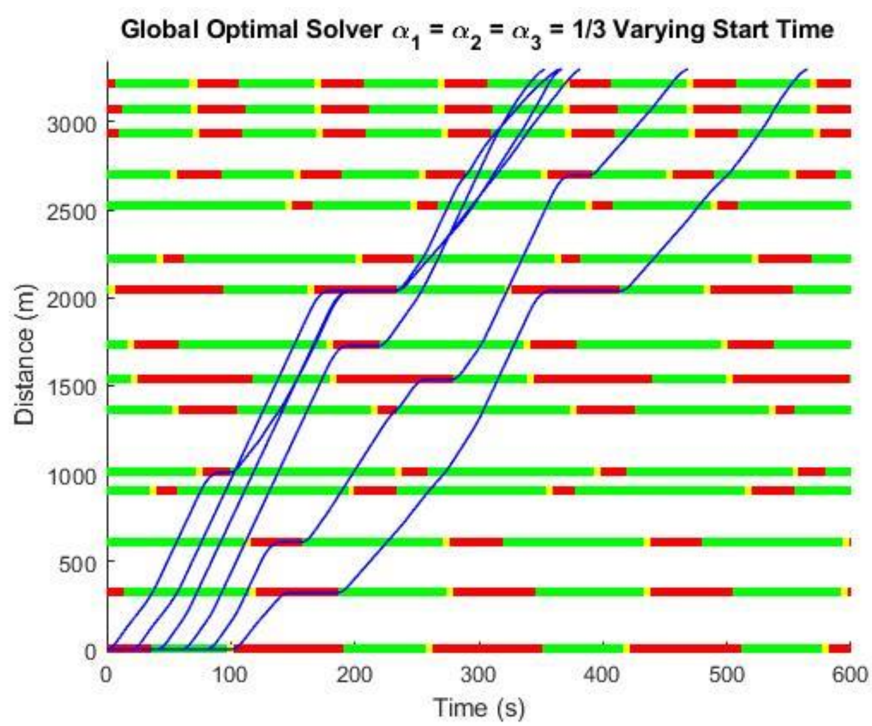

Figure 19 - Traversals with six varying start times for the Global Optimal Scheme

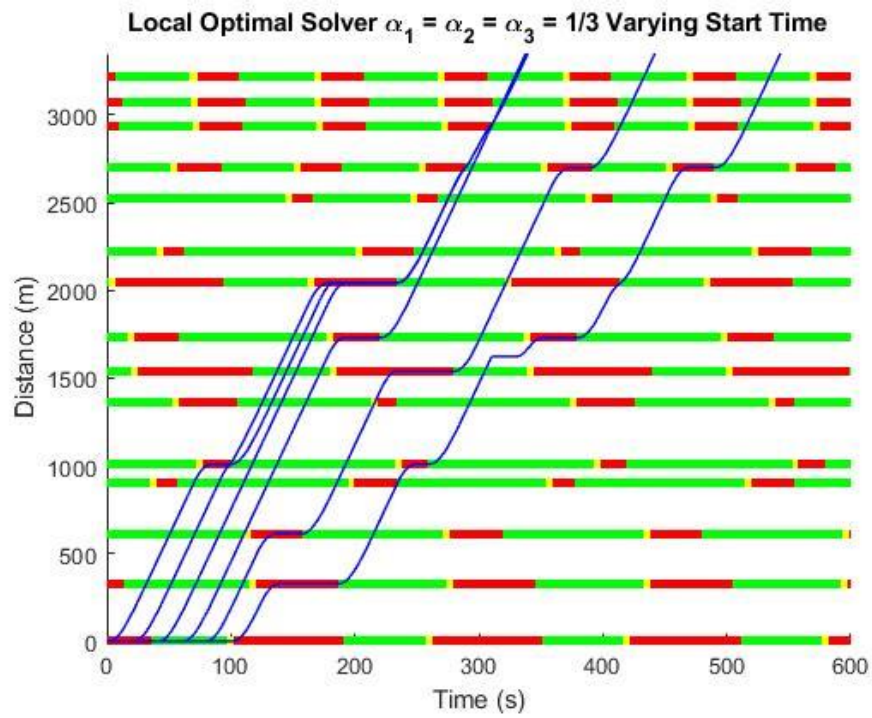

Figure 20 - Traversals with six varying start times for the Local Optimal Scheme

The Stop or Go Scheme performs the best in the traversal time metric, achieving a 12% improvement over the Global Optimal Scheme and a 5% improvement over the Local Optimal Scheme. With no penalty on acceleration or energy usage, traversals using the Stop or Go Scheme are characterized by an immediate acceleration to the user desired speed after coming to a stop at an intersection and maintaining this speed until a red light at a future intersection is encountered. The penalties placed on acceleration and energy usage in the Global Optimal and Local Optimal Schemes attenuate the acceleration and result in lower mean traversal times.

When the energy used cost is considered, the Global Optimal Scheme excels, performing 37% better than the Stop or Go scheme and 32% better than the Local Optimal Scheme. With complete knowledge of the corridor, energy usage is shown to be significantly reduced by optimizing the velocity trajectory over the corridor traversal. Similarly, the Global Optimal Scheme produces the lowest average acceleration across the six starting distances and the lowest standard deviation for this metric as well.

In the Stop or Go and Local Optimal Schemes, the traversals converge to a common path after coming to a stop at the same intersection. However, the Global Optimal Scheme paths do not converge despite stopping at the same intersection. This is expected since the Global Optimal Scheme considers the cost of past states when evaluating future states. For example, the time taken to reach an intersection will dominate the total cost if two controllers are initialized at the same intersection but different times.

Another interesting point in the responses is the tapering from the velocity trajectory in all responses of the Global Optimal Solver. This is because the search algorithm has advance knowledge of the final location. Whereas both the Stop or Go Scheme and the Local Optimal Solver only look at the current time step and the environment in a neighborhood of the vehicle, the Global Optimal Solver knows exactly where the vehicle needs to stop and can coast through the finish line. This response is analogous to a human letting their foot off the gas as the vehicle cruises through the end of a light.

2.4 Validation of results

2.4.1 Model-In-The-Loop Validation

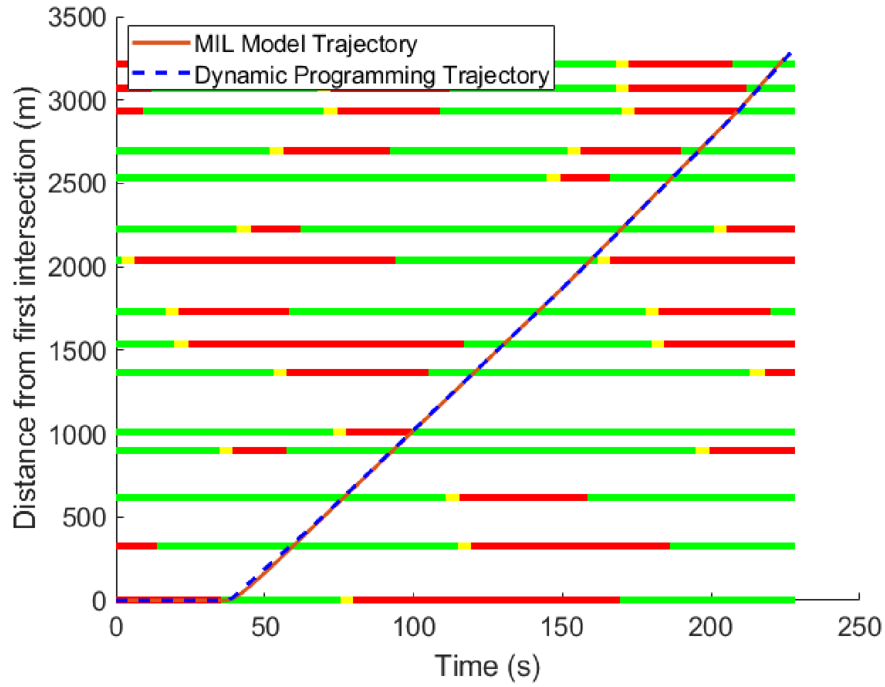

Figure 21 - Validation of most aggressive velocity trajectory in MIL environment

For the Global Optimal Scheme in the 40 mph speed limit case with the most aggressive acceleration, the velocity trajectory obtained through the search algorithm is used with the velocity regulator in the Model-In-the-Loop (MIL) model to validate that the trajectory is achievable with plant dynamics and actuator limits accounted for. In this simulation, the initial acceleration of the vehicle is not matched by the MIL model, leading to a maximum speed error of 3.37 m/s that is largely due to reaching the actuator limit of the fast-acting electric motor and the actuator delays expected from an internal combustion engine. After the initial acceleration event, the MIL model matches the velocity trace with an RMS error of 0.10 m/s, validating that this speed trace is implementable with vehicle dynamics considered.

2.4.2 Vehicle-In-The-Loop Testing on Ponce de Leon Avenue

To validate the simplifications made in both the vehicle models and simulation structures, the final component of this investigation was developed to test one of the schemes described above in the vehicle on the Ponce de Leon connected corridor. The first scheme, Stop or Go, was selected to be implemented in the vehicle, since it was the simplest, the team had already deployed something similar in a closed course environment, and it would be effective in analyzing the accuracy of the simulation environment. Additionally, experimental variables were identified ahead of time to be measured if one proved to be more impactful to the performance of the control strategy. These variables were road grade and lead vehicle longitudinal displacement data.

As a first step towards the implementation of this feature in a consumer vehicle, the team sought to validate the Stop or Go Scheme in a real-world setting. The Ponce de Leon Avenue connected corridor was traversed using this scheme as seen in Figure 22. The vehicle's ACC feature was active during this traversal and would be activated if a lead vehicle was detected, or no SPaT data was being received. SPaT data for this traversal was obtained using an API provided to Georgia Tech by the GDOT. The vehicle traverses the first intersection at t = 121 s.

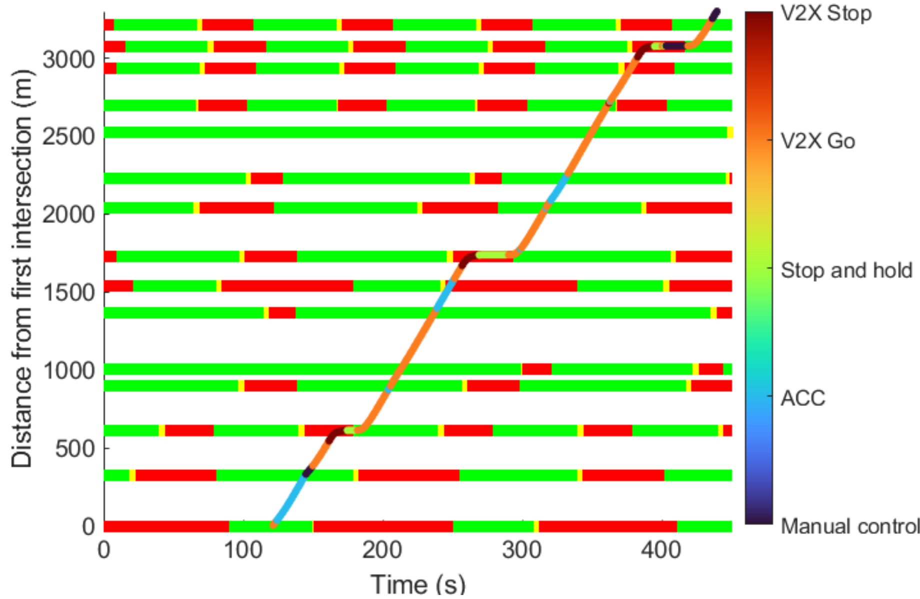

Figure 22 - Speed trace of successful Ponce de Leon connected corridor traversal with CAV operating modes highlighted

In this traversal, three autonomous stops were executed successfully without driver intervention, as seen by the three periods of rest. For each of these, V2X Stop mode was entered prior to engaging a "Stop and Hold" mode at the light, where the vehicle's Electronic Brake Control Module is commanded to hold the vehicle stationary. Figure 23 is a zoomed in view of Figure 22 when the vehicle traverses the intersection at d = 3078 m. Here, the vehicle correctly comes to a stop at an intersection, but due to noise in the GPS data and inaccurate intersection MAP data, proceeds to creep forward into the intersection in V2X Go mode. Driver intervention was necessary to return to manual control and prevent this.

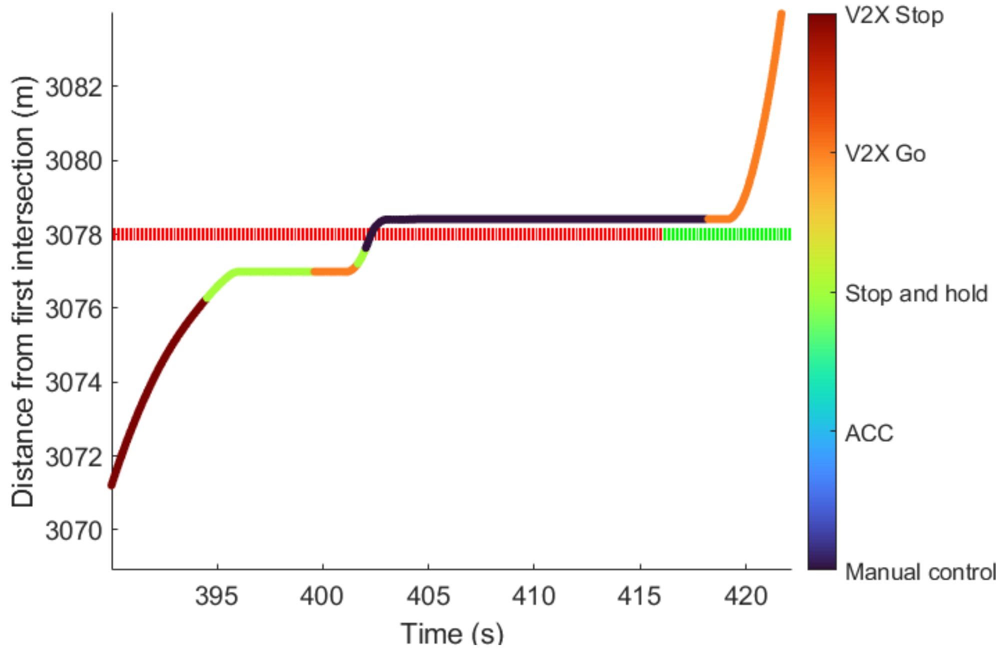

Figure 23 - GPS noise and inaccurate MAP data resulting in vehicle movement after initial stop

VIL testing revealed several limitations to real-world implementations of V2X algorithms. Inaccurate MAP data programmed into RSUs and noise in the GPS signal can lead to the vehicle coming to a stop beyond the stop line. Filtering of GPS noise and the visual identification of a stop line using cameras are possible approaches to solving this issue. Inaccurate SPaT data was also an issue. On some traversal attempts, an intersection with a green phase would broadcast a red phase with a time stamp outside the allowable range, indicating an undefined or unknown value. This would lead to undesirable behavior, including attempting to brake while traversing the intersection. Several intersections were not broadcasting SPaT and MAP data at all, presumably due to faulty RSUs. The team reached out to the GDOT and confirmed the team's findings that some of these intersections were not broadcasting information.

The presence of other vehicles is another confounding variable, prohibiting the ego vehicle from following a pre-defined optimal velocity trajectory if the exhaustive search algorithm approach is employed. In this traversal, there were 3 instances of entering ACC mode where a lead vehicle was identified. Finally, the road grade is not accounted for in simulation when this traversal includes 161 ft of elevation gain and 141 ft of elevation loss. The optimal velocity trajectory is affected by this variable, especially on steep gradients.

2.5 Summary and Future Work

2.5.1 Impact to the EcoCAR Mobility Challenge

The Connected Mobility Challenge at the EcoCAR Mobility Challenge Year 4 competition challenges teams to navigate a single signalized intersection broadcasting real-time SPaT and MAP data. Teams are scored on their ability to traverse the intersection correctly, by not running any red lights, and the stopping distance from the stop line in the event of a red phase. No lead vehicle will be present at this event and the vehicle will begin receiving SPaT and MAP data 160 m away from the intersection. Due to the nature of the competition requirements, the energy and comfort components of the cost function developed for the Global and Local Optimal Solver in this analysis are not necessary to be considered. Instead, two metrics take precedence to maximize points: having the correct response to a signal phase and minimizing the displacement error at the stop line.

The Stop or Go Scheme is the best choice for this competition event due to its simplicity, proven robustness through closed course and on-road testing, and track record of consistent performance at multiple testing events over the year. Two key learnings from this report will be applied in the competition event to maximize the team's performance.

In VIL testing, the only instance where the algorithm responded incorrectly to the signal phase was where the RSU broadcasted SPaT data outside the allowable range, indicating an undefined or unknown value. Previously, data from this undefined state was accepted as valid SPaT input, leading to controller behavior that violated the signal phase. This case has now been accounted for in the algorithm by ignoring undefined SPaT data until valid data is received from the RSU. This will aid the team in achieving the correct response to a signal phase 100% of the time.

VIL testing has highlighted the sensitivity of the stopping distance to MAP data and GPS noise. Several trials of this event have been done at the EcoCAR Spring testing event in March 2022 and an offset that minimizes the displacement error at the stop line has been calibrated. At the final competition, this offset will need to be validated to ensure maximum points are scored in this competition event.

2.5.2 Key Takeaways

The Georgia Tech team set out to develop a feature capable of automating the longitudinal motion of a vehicle to traverse a connected corridor and succeeded in developing three schemes to implement this feature. Each was validated in a simulation environment, and one was tested in a consumer vehicle on the streets of Atlanta.

The first scheme was derived from the work the team had already completed in the implementation of an ACC feature, where the vehicle needed to regulate velocity or distance to a lead vehicle depending on the environment. This scheme exhibited excellent time performance, running though the Ponce de Leon corridor consistently faster than the Global and Local Optimal Schemes.

The second scheme, the Global Optimal Scheme, sought to improve the feature's performance based on team defined cost functions. Armed with prior knowledge of all SPaT data, the best results of any scheme were achieved. This scheme illuminated interesting behavior abnormal to human behavior but serves as an excellent comparison for the best performance any scheme is capable of.

The final scheme, the Local Optimal Scheme, investigated a modification of the Global Optimal Scheme to make it implementable in a vehicle utilizing the team's current CAVs architecture. This scheme was able to demonstrate improvement over the Stop or Go Scheme with the same knowledge of the environment. This scheme exhibited the perfect blend of performance and practicality that the EcoCAR competition demands.

The impact of allowing speed limit violations was a faster traversal time but significantly worse energy and comfort costs. This result will need to be explored in future work by augmenting the searching algorithm or methodology used.

The relative performance of each scheme showed good robustness to different SPaT sequences. This validates the applicability of each scheme to the real world where connected corridors may have changing SPaT sequences throughout the day.

The Stop or Go Scheme was successfully integrated into the vehicle and tested on public roads. This surfaced several limitations around real-world functionality of this feature including the reliability of SPaT and MAP data from RSUs and the presence of traffic. These challenges will need to be addressed appropriately in future work to reduce their impact on algorithm performance.

2.5.3 Anticipated future investigation

The EcoCAR EV Challenge is the next installation of the AVTC series and is slated to begin in Fall 2022. This competition will have an expanded focus on connectivity and cooperative driving automation, necessitating the implementation and testing of V2X algorithms. The work done in this report will serve as a foundation for supporting development of these features.

Additionally, the team has outlined and developed the Local Optimal Scheme for implementing the connected corridor traversal feature, which performs in certain metrics much better than the traditional Stop or Go Scheme. In this paper, only 4 variations of the cost function weights are explored for this scheme. Customer preferences around time, energy and comfort can be explored to determine the combination of weights that will best appeal to the team's target market. The ability of a user to augment these weights on the fly can also be explored to meet different user needs.

Other future work around this scheme includes integration in a consumer vehicle, calibration to parameters do not present in the simulation environment, and further optimization of the algorithmic implementation. One such algorithmic improvement could be an intelligent state search algorithm. As mentioned, the current search algorithm performs a depth first search, but many algorithms exist, such as Dijkstra's algorithm and A* search, which could reduce the runtime of the search algorithm. A different algorithm might also overcome the affinity of the search algorithm to higher speeds even when they may be less optimal by exploring the search space more effectively.

Another area of future work anticipated in integrating a feature like this in real-world environment is the ability to account for other vehicles on the road. The team has currently developed an ACC feature and implemented a first pass of integrating it with the Stop or Go Scheme, where the ACC control request will be honored if a lead vehicle is too close, otherwise the Stop or Go scheme will take priority. Interesting edge cases were observed in the limited VIL testing performed on Ponce de Leon Avenue and open the door for much more rich future investigation. This integration with other automation features poses a unique challenge to schemes utilizing the search algorithm, which operates in an open loop manner, meaning that any interruption in its execution could lead to large deviations from desired behavior.

The evaluation in this paper was performed on a SPaT sequence obtained along the Ponce de Leon Avenue corridor. There are more than 1000 connected intersections in Atlanta [5], many along continuous corridors. These corridors may have different speed limits, levels of traffic, elevation changes and SPaT sequences that may pose different challenges that were not explored in this paper. As the EcoCAR competition increases its emphasis on V2X technology, the presence of this wealth of testing environments will give the Georgia Tech team ample opportunity to innovate and explore different traversal schemes in the next competition.

CHAPTER 3. DEVELOPMENT AND TESTING OF A HYBRID VEHICLE

3.1 Hybrid Supervisory Controller Development

A hybrid supervisory controller (HSC) is central to the team's development and integration of a hybrid vehicle in this competition. The EcoCAR competition specifies several Vehicle Development Process (VDP) goals at the beginning of the year to guide each team's development process. Several of these goals rely on a fully developed HSC, including having an advanced energy management strategy and fully-functional advanced fault detection & diagnostic strategies.

To achieve these goals, several functions of the supervisory controller exist such as:

- Control all team-added propulsion components
- Interface with Electronic Control Units (ECUs) that exist on the production vehicle
- Execute startup and shutdown sequences of components
- Provide feedback to the operator on system status
- Provide an interface for the Tank to achieve longitudinal control of the vehicle
- Arbitrate between the driver and Tank when autonomous features are in use

The Georgia Tech team uses a dSPACE MicroAutoBox II (MABX) as the HSC due to its significant automotive I/O capability and robustness as a rapid prototyping controller. The development of the HSC was started in Year 2, and since then has been a joint effort by several individuals on the Propulsion Controls and Modeling (PCM) sub-team over the past three years.

3.1.1 Supervisory Controller Architecture

3.1.1.1 Hardware Architecture

The controller hardware architecture is shown in Figure 24. Central to this architecture is the HSC which interacts with several production controllers indicated in white, other team-added controllers indicated in green, and third-party controllers on components that the team integrated onto the vehicle. Of note is the presence of three separate team-programmed gateways, each necessary to augment CAN communication to interface correctly with GM production controllers.

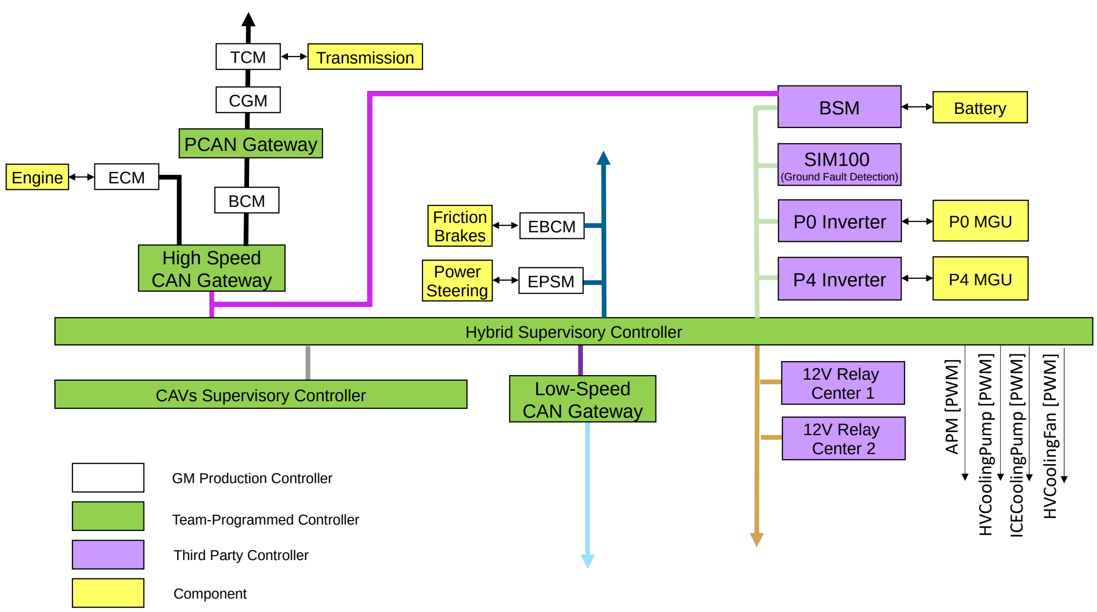

Figure 24 - Vehicle Controller Hardware Architecture

3.1.1.2 Software Architecture

The team implements algorithms for the HSC in Simulink. Simulink Code Generation is then used to generate the executable that is deployed onto the MABX. The HSC software architecture consists primarily of an input layer, application layer and an output layer. The input layer receives CAN, Digital and Analog data from other controllers in the vehicle using the MABX I/O capabilities and converts them into Simulink signals that can be used by the team's algorithms. The application layer contains all the team's algorithms, implemented in modular sub-systems. The output layer uses the MABX I/O capabilities to convert Simulink signals into CAN, Digital and Analog signals that are then transmitted to various components in the vehicle to achieve a desired objective.

3.1.1.3 Application layer

The design of the application layer makes use of a key principle of software architecture: partitioning. The application layer consists of upwards of 20 sub-systems as seen in Figure 25, each with its own unique function. The goal with this architecture is to enable distributed development, where members of the PCM sub-team can work on developing sub-systems in parallel. The coupling between modules is minimized as much as possible, to reduce interaction between sub-systems and improve the ability to understand code behavior.

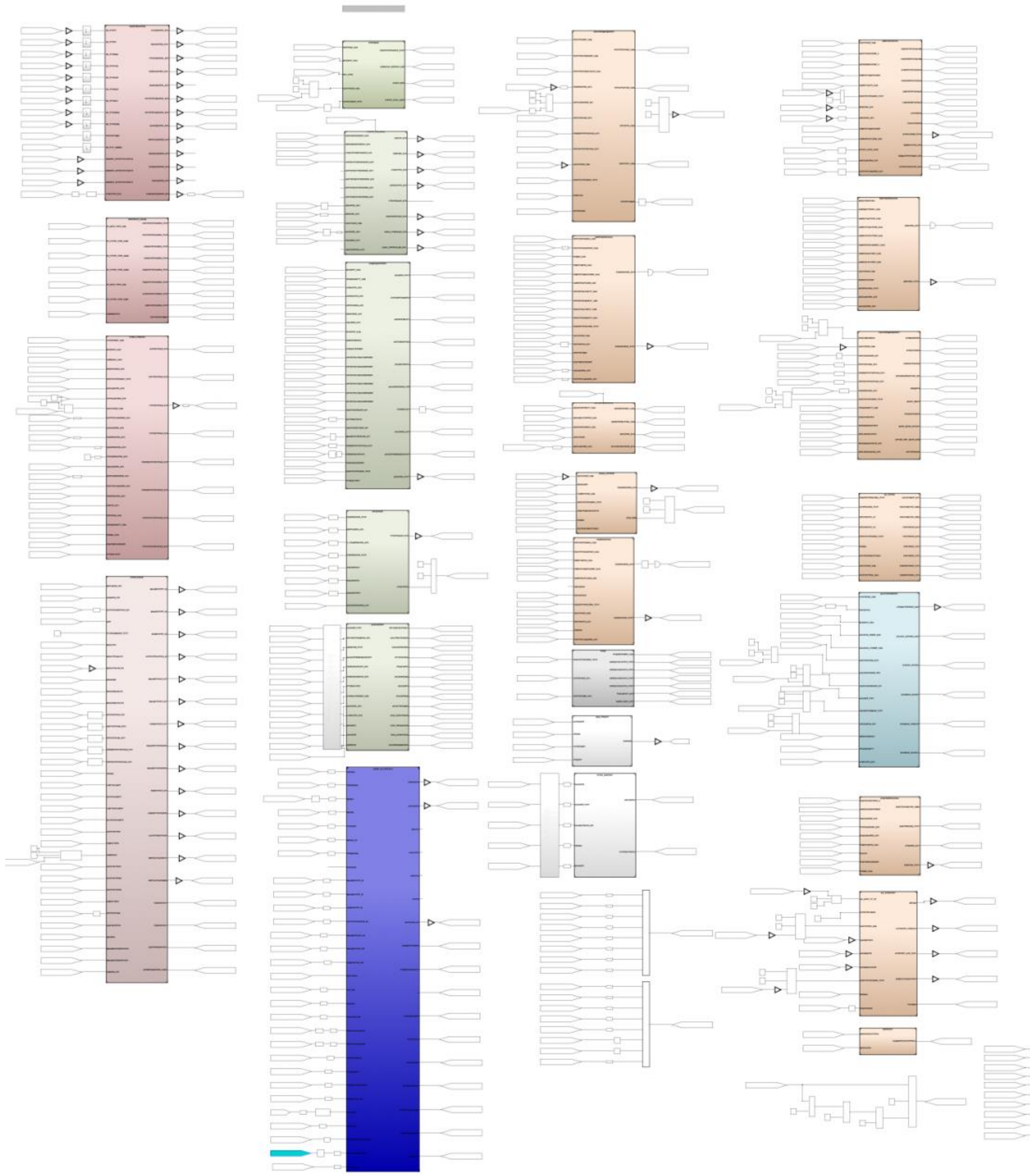

Figure 25 - HSC Application layer consisting of 26 sub-systems

3.1.2 Requirements development and testing

Each sub-system goes through a comprehensive requirements and test case development process before being tested in both MIL and VIL environments, as seen in Figure 26. After safety analysis is performed on the sub-system, safety requirements are delegated to sub-systems, and sub-systems are added if necessary to cover the scope of the safety requirements. All requirements and test cases are then documented systematically in sub-system documentation before MIL test cases are executed in the Simulink test environment, and VIL test cases are executed in the vehicle.

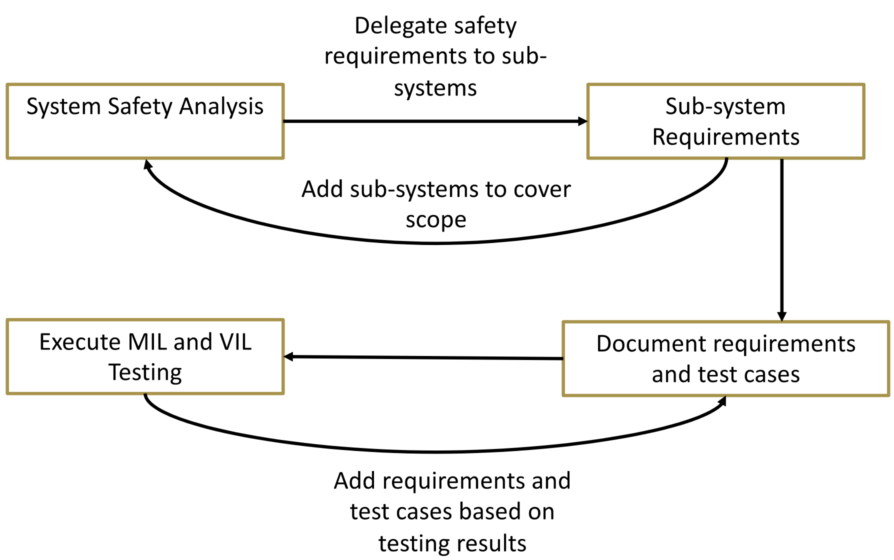

Figure 26 – Sub-system software development process

3.1.2.1 <u>Systems Safety Analysis</u>

The starting point for requirements development is understanding the functionality required of the system, followed by performing a safety analysis, identifying requirements, and finally decomposing these requirements into sub-systems. Several safety analyses were performed to align with GM’s best practices in Systems Safety that are applied in developing production vehicles. These analyses included Systems Theoretic Process Analysis (STPA), System Element Fault Analysis (SEFA) and System Functional Interface Analysis (SFIA). Based on these analyses, safety hazards are identified, from which safety requirements are derived and then allocated to sub-systems in the HSC. An example of such a breakdown is shown in Figure 27. In this example, a “Too Early” keyword from the SFIA analysis is used to identify the hazard of the P4 MGU producing torque while the vehicle is in Park, potentially damaging the MGU and the half-shafts that deliver torque from this MGU to the rear wheels. The safety requirement describes the need to set torque commands for the P4 MGU to 0 when the transmission is in Park. To implement this safety requirement, it is delegated to two existing sub-systems in the HSC, Torque Protection and P4 Inverter Management, where individual requirements are identified for each of those subsystems.

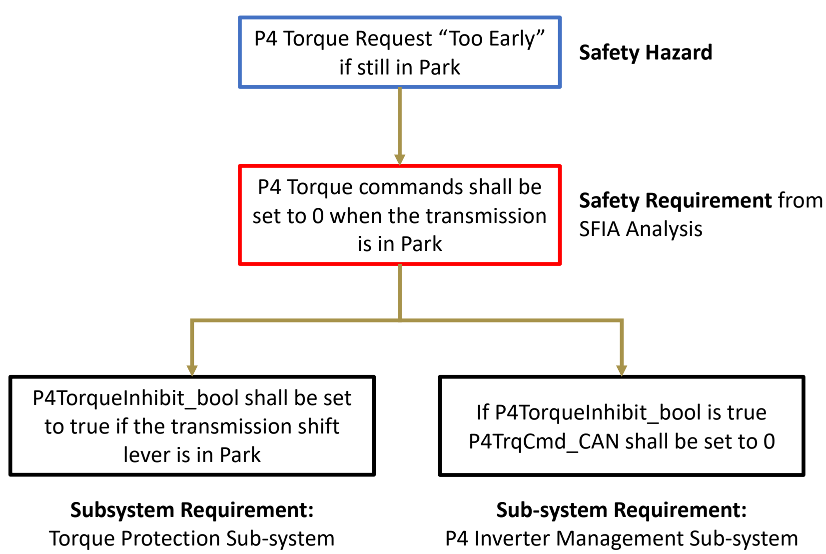

Figure 27 - Requirements development process from identification of safety hazard to sub-system requirements

Through this systematic process of developing safety requirements and decomposing them into sub-system requirements, the team has developed a robust HSC that is well-documented and can be leveraged by the team in the next competition, the EcoCAR EV Challenge, to accelerate the supervisory controller development process.

3.1.2.2 Requirements Documentation

Each of the sub-system requirements developed through the process described earlier are documented in one of several documents, each dedicated to a sub-system in the HSC. These documents include a change log, inputs, outputs, requirements, and test cases, providing a one-stop shop for all knowledge of a sub-system. In the EcoCAR EV Challenge, the format of these documents will be modified to include justifications for each requirement, thus improving the knowledge transfer from one year to the next.

3.1.2.3 MIL Testing

Test cases are developed for each requirement using a systematic process that considers every possibility and edge case. These test cases are then documented in the sub-system document described earlier. Execution of these test cases in the MIL environment is done using Simulink test harnesses. These test harnesses allow input signals to be manipulated for each sub-system and the outputs verified based on what the requirement is, and the condition being evaluated by the test case. A short section of the test harness for the Torque Protection subsystem is shown in Figure 28 and highlights how the requirement in Figure 27 is verified. In this case, the transmission shift lever position input is set to 3 to indicate that it is in Park. The sub-system output, P4TorqueInhibit_bool is then verified to be true and this verification statement is tagged with a test case ID that is used to update documentation. This process is used to verify all requirements developed by the team.

```
PCM_TP_2_5_2
endTest = false;

    Setup_1
    TransmissionShiftLeverPosition = 3; %Transmission in Park

    Verify_1
    verify(P4TorqueInhibit_bool  == true, 'ID:PCM_TP_2_5_2','fail');

    EndTestCases
    endTest = true;
```

Figure 28 - Test case example from Torque Protection sub-system test harness

3.2 Vehicle Testing Strategy

The development of a HSC with an advanced energy management strategy and fully-functional advanced fault detection & diagnostic strategies is just one aspect of the VDP goals in Year 4 of the competition. Other goals include a reliable propulsion system, good drive quality, optimized fuel economy, a full speed range Adaptive Cruise Control feature, and performing automated navigation of a V2X intersection. These goals are distributed across the 3 sub-teams, PCM, CAV, and Propulsion Systems Integration (PSI). Since time with the vehicle is a finite resource and each sub-team needs to execute testing towards meeting the VDP goals of the competition, a vehicle testing strategy was developed at the beginning of Year 4 to manage testing among all sub-teams. This would ensure each sub-team had the necessary time and resources allocated to perform testing, and any testing locations could be secured ahead of time if they were not immediately accessible to the team.

3.2.1 Vehicle validation goals

Vehicle testing activities were organized into 4 distinct validation goals in Year 4, as seen in Table 11. Each of these validation goals have associated VDP goals, VTS targets or competition events that guide the testing activities towards achieving it.

These 4 validation goals are chosen to distinguish the approaches or requirements for each testing day throughout the year. Propulsion System Functionality is a critical validation goal that enables the other three validation goals. This is focused on developing a robust and reliable vehicle that can be driven consistently and has maximum up-time. ACC functionality is an important competition requirement and all activities related to the perception system and longitudinal controller fall under this validation goal. Vehicle Performance Specifications are metrics that the team needs to meet to succeed in the competition. Besides minimum requirements like acceleration and braking, drive quality and energy consumption will be tuned throughout the year to achieve this validation goal. Finally, with a focus on the Connected Mobility Challenge at the Year 4 competition, where the vehicle will autonomously navigate a connected intersection, a fourth validation goal is specified for all development and testing activities geared towards V2X functionality.

Table 11 - Summary of vehicle validation goals

Validations Goal	Associated Needs from VDP Goals, VTS Targets, Testing Goals or Competition Events
Propulsion System Functionality	• 1500 accumulated miles • 200-mile continuous drive • Advanced fault detection and diagnostic strategies • Robust component operation
ACC Functionality	• Reliable perception system • Reliable longitudinal controller • 400 miles of longitudinal control testing • 60-mile continuous ACC drive
Vehicle Performance Specifications	• Meet drive quality metrics • Meet acceleration targets • Meet braking targets • Meet vehicle fuel economy target through efficient energy management strategy
V2X Functionality	• Connected Mobility Challenge

3.2.2 Vehicle test plan

To accomplish these validation goals, a detailed test plan was developed, a portion of which is visualized in Figure 29.

	January		February			
	17-Jan	24-Jan	31-Jan	7-Feb	14-Feb	21-Feb
Propulsion System Functionality	Mileage accumulation for self-certification on Version 4.00	Regenerative Braking Slope Testing	P0 Usage Improvements	Coast regenerative braking	Generation Mode Testing	Final testing and debugging on version 5.00
ACC Functionality	ACC Braking	Lane Detection verification Data collection	Sensor fusion validation testing & data collection	100-mile ACC Endurance Drive, Baseline Fuel Economy	Sensor fusion validation testing & data collection	Sensor fusion validation testing & data collection
Vehicle Performance Specifications	-	Run drive cycle on v4	Re-run drive quality events	-	-	-
V2X Functionality	-	-	-	Curiosity Labs V2X Testing	Curiosity Labs V2X Testing	-
PropSys Miles Target	700	850	900	1030	1080	1100
PropSys Miles Over/Under	170	248	198	TBD	TBD	TBD
CAV Miles Target	120	170	210	310	350	390
CAV Miles Over/Under	28	50	11	TBD	TBD	TBD

Figure 29 - Excerpt of vehicle test plan

This was a living document that was updated throughout the year based on evolving circumstances. For example, when transmission damage was sustained on the vehicle in December 2021, adjustments were made to the schedule as seen in Figure 30.

	December	January	
	13-Dec	17-Jan	24-Jan
Initial Vehicle Performance Specifications Schedule	Complete drive cycle	-	-
Revised Vehicle Performance Specifications Schedule	- 2-week delay	-	Complete drive cycle

Figure 30 - Schedule adjustment due to transmission damage in December

The planning, maintenance, and execution of this test plan was critical to the team's success and allowed the team to exceed the competition mileage targets for both the propulsion system and longitudinal control system three months before the beginning of competition. This paved the way for reduced testing load towards the end of the semester, reducing the risk of damaging critical components right before the competition.

3.2.3 Data processing, storage, and visualization

Completing all the testing activities in the test plan generated large amounts of data from the data logger that is integrated into the vehicle. A systematic approach to storing and visualizing this data was necessary to ensure that key component operation metrics were being monitored and that component faults were being detected and addressed. At the end of each testing day, data logs were extracted from the vehicle data logger into a hard drive and uploaded to a cloud storage folder for long term storage. These logs were then fed into a MATLAB script that would parse all log files and automatically produce several useful outputs:

1. A PowerPoint presentation showcasing various plots of component operation and any faults detected
2. A text file summarizing all the faults detected, and the time stamp in a particular log file that the fault occurred.
3. An Excel file summarizing key data from each individual log file for ease of understanding the utility of each log file.
4. A .mat file containing uniformly-spaced data streams from 143 key parameters on the vehicle for the whole duration of the log files, including component temperatures, current, voltage, torque and speed, and diagnostic signals.

The .mat file can be used for any data processing or plot generation in MATLAB. This makes data accessible to any individual on the team in an easy-to-use format to generate visualizations or diagnose component behavior. Generation of the PowerPoint presentation uses MATLAB's API for PowerPoint and automatically produces a 43-slide presentation that visualizes the behavior of the engine, P4 MGU, P0 MGU, 12V system, High Voltage Cooling System during the period of the log files. These visualizations allow the team to validate component functionality quickly and provides a consistent means to evaluate the performance of various components daily.

3.3 Summary and Future Work

In Year 4 of the EcoCAR Mobility Challenge, a reliable powertrain, ACC feature, and connected corridor traversal feature were developed. These features have been tested on public roads in Atlanta and have been validated to meet team requirements, culminating in an autonomous traversal of the Ponce de Leon Avenue connected corridor.

A rigorous development and testing process was followed in Year 4 to enable this success, including a well-structured test plan that enabled the successful completion of each validation goal, and a thorough requirements development and testing process that resulted in a HSC that meets the competition VDP goals.

The systems and processes described in this thesis will be built on in the next competition to further enhance the efficiency of the team's operation. The format of sub-system documentation is an area that deserves attention. A documentation system that captures more insight into the thought process behind requirements and test cases needs to be developed. This will not only improve the ability to on-board new members on a sub-system they are assigned to work on, but will better capture key learnings as team members change from one year to another. Opportunities to integrate this documentation system with

Simulink's environment using tools such as Simulink Requirements and Simulink Test should be explored, to improve the efficiency of the team's operations.

While the first year of the EcoCAR EV Challenge will preclude vehicle testing activities, the data processing and visualization workflow developed in this competition is a good foundation for future work in this area. In the second year of the EV Challenge, once a vehicle is integrated and testing activities have commenced, more visualizations should be explored, and the format and presentation of data should evolve based on the architecture of the next vehicle.

REFERENCES

[1] Georgia Department of Transportation, "Georgia Department of Transportation: Road & Traffic Data," [Online]. Available: http://www.dot.ga.gov/DS/Data. [Accessed 5 April 2022].

[2] I. Bae, J. Moon and S. J. , "Toward a Comfortable Driving Experience for a Self-Driving Shuttle Bus," *Electronics*, vol. 8, no. 9, p. 943, 2019.

[3] A. Kilinc and T. Baybura, "Determination of minimum horizontal curve radius used in the design of transportation structures, depending on the limit value of comfort criterion lateral jerk," Rome, 2012.

[4] *Code of Federal Regulations Title 47 Part 90 Private Land Mobile Radio Services.*

[5] A. Davis, "Georgia Connected Vehicles: Progress and Plans," Georgia Department of Transportation, Atlanta, 2019.

[6] Cohda Wireless, "Cohda Wireless: MK5 OBU," [Online]. Available: https://www.cohdawireless.com/solutions/hardware/mk5-obu/. [Accessed 7 April 2022].

[7] F. Bai and H. Krishnan, "Reliability Analysis of DSRC Wireless Communication for Vehicle Safety Applications," *IEEE Intelligent Transportation Systems Conference*, pp. 355-362, 2006.