PLATFORM DESIGN FOR CUSTOMIZABLE PRODUCTS AND PROCESSES WITH NON-UNIFORM DEMAND

A Thesis
Presented to
The Academic Faculty

by

Christopher Bryant Williams

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

Georgia Institute of Technology
September, 2003

PLATFORM DESIGN FOR CUSTOMIZABLE PRODUCTS AND PROCESSES WITH NON-UNIFORM DEMAND

Approved by:

Dr. Farrokh Mistree, Co-Chair
Professor
Mechanical Engineering

Dr. David W. Rosen, Co-Chair
Associate Professor
Mechanical Engineering

Dr. Janet K. Allen
Senior Research Scientist
Mechanical Engineering

Dr. Paul M. Griffin
Associate Professor
Industrial and Systems Engineering

Dr. Marie C. Thursby
Professor
Management

Date Approved November 24, 2003

ACKNOWLEDGEMENTS

When I first visited the Systems Realization Laboratory at Georgia Tech, I was immediately ready to join. The open environment, the non-competitive atmosphere, and the outgoing students and faculty really created the sort of "graduate family" that I wanted to be a part of. The beauty of being a part of this collection of graduate students is the opportunities I have had to take advantage of the diversity in culture, opinions, and problem solving techniques present in the lab. Throughout the completion of this thesis I have worked with many colleagues, many advisors, and most importantly, many friends.

First and foremost I must acknowledge my co-advisors, Dr. Farrokh Mistree and Dr. David Rosen. I thank Farrokh for providing me motivation and inspiration; for helping me realize (and hopefully surpass) my potential; for being genuinely excited about, and helping me achieve, my dream of being a professor; for keeping an open door; and most importantly, for lighting a fire under me when I needed it most. I thank Dave for providing concrete research advice; for providing an honest opinion; for providing amazingly insightful comments; and, of course, being concerned about my progress when I needed it most. I consider myself to be extremely lucky to have such amazing research advisors – although they are extremely opposite in advising approaches, there is one aspect that they share in common: the desire to see their students succeed.

I also must thank my reading committee: Dr. Janet Allen, Dr. Paul Griffin, and Dr. Marie Thursby. All three provided unique insights and helpful questions and comments to strengthen this body of work.

In addition to these "professional" advisors, I have had the pleasure with working with several students who also provided a lot of quality advice. Specifically, I would like to thank Gabriel Hernandez for leaving behind such an amazing research legacy as the PPCTM. During his last months with the Systems Realization Laboratory, Gabriel served as my mentor; he advised me, gave me direction, and introduced me to his research. Perhaps more important to the completion of this thesis, is the manner in which Gabriel continued to provide guidance after his graduation. Gabriel's passion for his research extended far beyond his new home in Orlando as he and I continued to corresponded through e-mail and telephone. From Gabriel I have learned the importance in leaving a research legacy upon graduation.

I also owe a large amount of gratitude to Carolyn Conner-Seepersad. Carolyn was an extra, behind-the-scenes advisor throughout my research. Not only did she provide numerous lessons on utility theory, but she also joined me in several brainstorming sessions even though they weren't directly related to her work. Carolyn also provided inspiration on how to effectively validate my research. The aspect that I am the most thankful for is how she continued to smile despite my countless interruptions. Carolyn is one of the most amazing students I have ever met – she will be an incredible professor and a terrific advisor: she already has years of experience.

I must also thank my colleague Jitesh Panchal for working with me on the hearing aid shell example problem. I am grateful for the utility theory lessons and JAVA programming tips that Jitesh provided me. I also acknowledge the teamwork of Jae Park, Ryan Johnson, Jitesh and myself, as we developed the model for the hearing aid shell together as a team.

I thank Matthew Chamberlain for keeping me entertained, providing me with thesis advice, and keeping me focused. I also thank Matt for being my doppelganger, my roommate, and my friend. Most importantly, I thank Matt for deciding to stay in the SRL in order to complete his PhD; his absence would have made my last years in the lab much less eventful.

I also offer my gratitude to Marco Fernández for being such a generous and thoughtful friend. Marco listened to my complaints and worries; but managed to keep me in the lab, to keep me working, and to keep my head up at all times.

Many other members of the lab offered help in the completion of this thesis. Adam Coutee provided the inspiration for the cantilever beam example. Benay Sager provided me with advice on the thesis and the presentation. Melissa Bargmann gave me countless MS Office lessons and shortcuts, and lots of laughs. I am grateful to them and to the lab as a whole for providing me with a wonderful working environment.

I also gratefully acknowledge the support of a G.W. Woodruff Presidential Fellowship and NSF Grants DMI-0085136, DMI-9612327, and DMI-9618039.

Finally, and most importantly, I must offer my gratitude to my family. My mother and father have always provided me with support and encouraging words throughout my education. They have instilled the importance of education in me, and for that I am grateful. I thank my sister for taking care of my dog while I was away, for offering words of encouragement, and for commiserating with me as we trudge thorough graduate school together. Of course I must also thank my in-laws-to-be for providing an additional home, an additional warm meal, and an additional set of wonderful role models.

Perhaps the most amount of gratitude that I owe belongs to my fiancée Justine Brantley. Without her support, love, understanding, and smile, I would not have been able to complete this thesis. She watched over me, worried about me, and always provided comfort when I needed it most. I am extremely lucky to have such an amazing and caring partner throughout the rest of my life.

TABLE OF CONTENTS

ACKNOWLEDGMENT iii

TABLE OF CONTENTS vii

LIST OF TABLES xv

LIST OF FIGURES xvii

LIST OF BOXES xxiii

GLOSSARY xxv

NOMENCLATURE xxviii

SUMMARY xxxii

CHAPTER 1 THE CHALLENGES OF DESIGNING FOR MASS CUSTOMIZATION **1**

1.1 BACKGROUND AND MOTIVATION FOR STUDYING MASS CUSTOMIZATION 2

1.1.1 The Contemporary Competitive Advantage: Product Variety 2

1.1.2 The Challenges of Mass Customization 5

1.2 THE NEED FOR SYSTEMATIC PRODUCT PLATFORM DESIGN METHODS 8

1.2.1 Bottom-up Approaches 10

1.2.2 Top-down Approaches 11

1.2.3 Limitations of Existing Top-Down Methods 11

1.2.4 Product Platform Design as a Problem of Access in a Geometric Space 13

1.3 RESEARCH FOCUS, QUESTIONS, AND HYPOTHESES 13

1.4 VALIDATION AND VERIFICATION STRATEGY FOR THIS THESIS 17

1.4.1 Theoretical Structural Validity 20

1.4.2 Empirical Structural Validity 20

1.4.3 Empirical Performance Validity 21

1.4.4 Empirical Structural Validity 21

1.5 ORGANIZATION OF THIS THESIS 21

CHAPTER 2 THE THEORETICAL FOUNDATION: AUGMENTING THE PRODUCT PLATFORM CONSTRUCTAL THEORY METHOD 27

2.1 PRODUCT PLATFORM DESIGN AS A PROBLEM OF ACCESS IN A GEOMETRIC SPACE 29

2.1.1 Theoretic Underpinnings: Hierarchic Systems Theory and Constructal Theory 29

2.1.2 The Product Platform Constructal Theory Method 35

2.1.3 Critical Evaluation of the Product Platform Constructal Theory Method 53

2.2 THE UTILITY-BASED COMPROMISE DECISION SUPPORT PROBLEM 56

2.2.1 Compromise Decision Support Problems 56

2.2.2 Utility Theory 57

2.2.3 Utility-Based Decision Support Problem 58

2.2.4 Critical Evaluation of the Utility-Based Compromise Decision Support Problem 67

2.3 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD 67

CHAPTER 3 A METHOD FOR DESIGNING PRODUCT AND PROCESS PLATFORMS AS A PROBLEM OF ACCESS IN A GEOMETRIC SPACE OF NON-UNIFORM DEMAND 71

3.1 THE MODELING OF NON-UNIFORM DEMAND 74

3.1.1 Discrete Function 75

3.1.2 Linear Continuous Function 77

3.1.3 n-Dimensional Gaussian Distribution 78

3.2 ABSTRACTING THE PRODUCT PLATFORM CONSTRUCTAL THEORY METHOD TO PROCESS DESIGN FOR CUSTOMIZED PRODCUTS 81

3.2.1 Definition of Process Platform 82

3.2.2 Definition of Space of Capacity 85

3.2.3 Definition of Modes for Managing Process Variety 87

3.2.4 Design of Process Platforms for Customizable Products as an Access Problem 89

3.3 COPING WITH MULTIPLE OBJECTIVES WITH THE UTILITY-BASED COMPROMISE DECISION SUPPORT PROBLEM 92

3.3.1 The Infusion of the u-cDSP into the PPCTM 92

3.3.2 A Critical Analysis of the Infusion of the u-CDSP into the PPCTM 97

3.4 A METHOD FOR DESIGNING PRODUCT AND PROCESS PLATFORMS IN A GEOMETRIC SPACE OF NON-UNIFORM DEMAND 98

3.4.1 Step 1: Define the Geometric Space and Demand Scenario 100

3.4.2 Step 2: Define the Objective Functions 102

3.4.3 Step 3: Identify the Modes for Managing Variety 103

3.4.4 Step 4: Identify the Number of Hierarchy Levels and Allocate the Modes for Managing Variety to the Levels 104

3.4.5 Step 5: Formulate a Multi-Stage Utility-Based Compromise Decision Support Problem 104

3.4.6 Step 6: Solve the Multi-Stage Utility-Based Compromise Decision Support Problem 106

3.5 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD 111

CHAPTER 4 CUSTOMIZED PRODUCT PLATFORM DESIGN: PRESSURE VESSELS 115

4.1 THE PROBLEM OF OFFERING CUSTOMIZED PRESSURE VESSELS IN A MARKET OF NON-UNIFORM DEMAND 117

4.1.1 Pressure Vessel Market Description 118

4.1.2 Pressure Vessel Product Model 119

4.2 EXPRESSING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE 123

4.2.1 Step 1: Define the Geometric Space and the Demand Scenario 124

4.2.2 Step 2: Define the Objective Functions 126

4.2.3 Step 3: Identify the Modes for Managing Variety 126

4.2.4 Step 4: Identify the Number of Hierarchy Levels and Allocate the Modes for Managing Variety to the Levels 128

4.2.5 Step 5: Formulate a Multi-Stage Utility-Based Compromise Decision Support Problem 132

4.2.6 Step 6: Solve the Multi-Stage Utility-Based Compromise Decision Support Problem 143

4.3 SOLVING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE 147

4.3.1 Discretization of the Geometric Space 147

4.3.2 Performing an Exhaustive Search 149

4.3.3 Modeling Demand 150

4.3.4 Evaluating the Results 157

4.3.5 Analysis of the Results 164

4.4 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD 170

CHAPTER 5 CUSTOMIZED PROCESS PLATFORM DESIGN: THE RAPID MANUFACTURING OF HEARING AID SHELLS 173

5.1 THE PROBLEM OF PRODUCING CUSTOMIZED HEARING AID SHELLS IN A MARKET OF NON-UNIFORM DEMAND 176

5.1.1 Hearing Aid Shell Market Description 179

5.1.2 Hearing Aid Shell Process Model 180

5.2 EXPRESSING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE 189

5.2.1 Step 1: Define the Geometric Space and the Demand Scenario 190

5.2.2 Step 2: Define the Objective Functions 191

5.2.3 Step 3: Identify the Modes for Managing Variety 193

5.2.4 Step 4: Identify the Number of Hierarchy Levels and Allocate the Modes for Managing Variety to the Levels 196

5.2.5 Step 5: Formulate a Multi-Stage Utility-Based Compromise Decision Support Problem 202

5.2.6 Step 6: Solve the Multi-Stage Utility-Based Compromise Decision Support Problem 217

5.3 SOLVING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE 222

5.3.1 Discretization of the Geometric Space 222

5.3.2 Applying a Gradient-Based Solution Algorithm 224

5.3.3 Evaluating the Results 226

5.3.4 Verification of Results 232

5.3.5 Analysis of the Results 236

5.4 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD 245

CHAPTER 6 A LOOK BACK AND A LOOK AHEAD: CLOSING REMARKS 248

6.1 A LOOK BACK: A SUMMARY OF WHAT HAS BEEN PRESENTED AND ITS CONTRIBUTION TO VALIDATION 250

6.1.1 Revisiting the Research Questions and Hypotheses 250

6.1.2 Testing the Validity of the Proposed Augmented Method 251

6.2 ACHIEVEMENTS AND CONTRIBUTIONS 261

6.3 A CRITICAL ANALYSIS AND REVIEW OF THE LIMITATIONS OF THIS RESEARCH 264

6.4 A LOOK AHEAD: FUTURE WORK 268

6.5 CLOSING REMARKS 270

APPENDIX A JAVA CODE FOR THE PRESSURE VESSEL EXAMPLE PROBLEM 276

A.1 LIST OF CLASS FILES AND FUNCTIONS 276

A.1.1 List of Class Files 276

A.1.2 List of Functions 277

A.2 SOFTWARE CODE 279

APPENDIX B MAPPING PRODUCT SPECIFICATIONS ONTO THE SPACE OF CUSTOMIZATION FOR EACH DEMAND SCENARIO OF THE PRESSURE VESSEL EXAMPLE 303

B.1 DISCRETE PYRAMID SMALL 304

B.2 LINEAR LARGE 306

B.3 LINEAR SMALL 314

B.4 NORMAL CENTER SMALL 314

B.5 NORMAL LOW 321

B.6 NORMAL HIGH 324

B.7 NORMAL CENTER LARGE 328

B.8 RANDOM LARGE 332

B.P RANDOM SMALL 340

APPENDIX C JAVA CODE FOR THE HEARING AID EXAMPLE PROBLEM 342

C.1 LIST OF CLASS FILES AND FUNCTIONS 342

C.1.1 List of Class Files 342

C.1.2 List of Functions 344

C.2 SOFTWARE CODE 346

APPENDIX D SOLUTION CONVERGENCE HISTORY OF HEARING AID EXAMPLE DATA FOR EACH MACHINE TYPE 373

D.1 ONE FDM MAXUM 373

D.2 TWO FDM MAXUMS 375

D.3 ONE FDM TITAN 377

D.4 TWO FDM TITANS 379

D.5 ONE PRODIGY PLUS 380

D.6 TWO PRODIGY PLUS 382

APPENDIX E MAPPING PRODUCT SPECIFICATIONS ONTO THE SPACE OF CUSTOMIZATION FOR EACH DEMAND SCNEARIO OF THE PRESSURE VESSEL EXAMPLE 384

E.1 TWO FDM MAXUMS 385

E.2 ONE FDM TITAN 387

E.3 TWO FDM TITANS 388

E.4 ONE PRODIGY PLUS 390

E.5 TWO PRODIGY PLUS 391

APPENDIX F EXPECTED UTILITY COMPARISION AMONGST MACHINES 393

REFERENCES 396

LIST OF TABLES

Table 1.1	The Differences Between Mass Production and Mass Customization (Pine, 1993)	6
Table 1.2	Relevance of Thesis Chapters and Sections to Research Questions and Hypotheses	16
Table 2.1	The Analogy of Creating a Street Network and Designing Product Platforms via the Application of Constructal Theory	36
Table 2.2	Illustrative Results of Cantilever Beam Tutorial Example	50
Table 2.3	Mapping Modes of Managing Variety to Product Specifications	52
Table 2.4	Cantilever Beam Utility Function Assessment	63
Table 3.1	Cantilever Beam Normal Distribution Parameters	80
Table 4.1	Designer Preferences for Profit of Pressure Vessel Example	133
Table 4.2	Decision Variables of Each Stage of Pressure Vessel Analysis	141
Table 4.3	Example Values for Modes of Managing Product Customization	145
Table 4.4	The Ten Demand Scenarios for the Pressure Vessel Example Problem	151
Table 4.5	Results for Demand Scenario#1, "Discrete Pyramid Large"	158
Table 4.6	Mapping Between Product Specifications and Design Variables for Design Scenario #1, "Discrete Pyramid Large"	159
Table 4.7	Range of Commonality of the Modes for Managing Product Customization for Each Demand Scenario of the Pressure Vessel Example	163
Table 4.8	Checking Constraints for Pressure Vessel Example	164
Table 5.1	Hearing Aid Shell Average Dimensions (Williams, et al., 2003)	181
Table 5.2	Machine Process Characteristics (Stratasys, 2003)	182

Table 5.3	Utility Functions for Cost, Time, and Quality	203
Table 5.4	Decision Variables of Each Stage of the Hearing Aid Analysis	216
Table 5.5	Example Values of Managing Process Variety	219
Table 5.6	Example Results for Illustration of Decision 2	221
Table 5.7	Sequential Quadratic Programming Application Conditions	225
Table 5.8	Range of Commonality of the Modes for Managing Process Variety for Each Machine Combination of the Hearing Aid Example	227
Table 5.9	A Sample of the Mapping Between the Process Platform and Parameters for One FDM Maxum	229
Table 5.10	A Sample of the Mapping Between Process Platform and Parameters for One Prodigy Plus	230
Table 5.11	Range of Each Mode for Managing Process Variety for Hearing Aid Example	231
Table 5.12	Results of SQP from One Maxum; Starting Point $\Delta D_1 = 5$, $\Delta D_2 = 50$, $\Delta D_3 = 150$	233
Table 5.13	Results of GRG from One FDM Maxum; Starting Point $\Delta D_1 = 5$, $\Delta D_2 = 50$, $\Delta D_3 = 150$	234
Table 5.14	Comparing Results of the SQP with an Exhaustive Search	235
Table 5.15	Similar Expected Utilities for Different Ranges of Modes for Managing Variety in Hearing Aid Example Problem	240
Table 5.16	Comparison of a) Augmented PPCTM Process Platform, b) No Commonality, and c) Strict Commonality of Process Parameters	244

LIST OF FIGURES

Figure 1.1	Number of Car Variants at BMW from 1975 to the Early 1990s (Ericsson et al., 1999)	3
Figure 1.2	Anybody, Anything, Anywhere, Anytime, Any Volume: The Five Goals of Customized Manufacturing (myb2o.com, 2001)	4
Figure 1.3	The Economic Implications of Mass Customization (Tseng et al., 1998)	5
Figure 1.4	The cost of Variety as a Function of market Variety (Anderson, 1997)	7
Figure 1.5	Average Life Cycle of Philips Electronic Media (Minderhoud, 2003)	9
Figure 1.6	Representation of the Research Thrusts for this Thesis	15
Figure 1.7	The Validation Square: Validating Design Theories or Methods (Penderson et al. 2000)	18
Figure 1.8	Validation Strategy for this Thesis	19
Figure 1.9	A Roadmap for this Thesis	22
Figure 1.10	The Cantilever Beam Tutorial Example Running Icon	26
Figure 2.1	The Role of Chapter 2 in this Thesis	28
Figure 2.2	Hierarchic (A) vs. Non-Hierarchic (B) Organization of Systems	30
Figure 2.3	A Finite-Size Area with a Common Destination (Bejan, 1997)	31
Figure 2.4	Organization in Nature as a Problem of Access Optimization (Bejan, 2000)	32
Figure 2.5	The First Area Element (Bejan, 1997)	33
Figure 2.6	The Second Area Element (Bejan, 1997)	34
Figure 2.7	Product Platform Design as a Problem of Optimal Access	35

Figure 2.8	The Use of Constructal Theory in Product Platform Design	37
Figure 2.9	Hernandez's Method for Designing Product Platforms for Customizable Products (Hernandez, 2001)	38
Figure 2.10	Customizable Cantilever Beam	38
Figure 2.11	Space of Customization for Cantilever Beam Example	39
Figure 2.12	Baseline Decision for an Arbitrary Stage *i* (Hernandez, 2001)	43
Figure 2.13	Baseline Decision for Stage 1 of Cantilever Beam Example	44
Figure 2.14	Baseline Decision for Stage 2 of Cantilever Beam Example	44
Figure 2.15	Baseline Decision for Stage 3 of Cantilever Beam Example	45
Figure 2.16	Formulation of the Multistage Problem as a Dynamic Program (Hernandez, 2001)	46
Figure 2.17	The Multistage Problem for the Cantilever Beam Example	47
Figure 2.18	Developing Response Surface Approximations for Optimal Objective Functions in Dynamic Programming (Hernandez, 2001)	48
Figure 2.19	The Solution Algorithm for the Cantilever Beam Example	50
Figure 2.20	Sample Hierarchic Solution of the Cantilever Beam Example	51
Figure 2.21	Hierarchical Arrangement of the Modes of Managing Product variety for the Cantilever Beam Tutorial Example	51
Figure 2.22	Augmenting the Compromise DSP with Utility Theory to Form the Utility-Based Compromise DSP (Seepersad, 2001)	58
Figure 2.23	Mathematical Formulation of the Utility-Based Compromise Decision Support Problem (Seepersad, 2001)	59
Figure 2.24	Monotonic and Non-Monotonic Utility Functions (Fernández et al., 2001)	60
Figure 2.25	Risk Averse vs. Risk Prone Utility Functions (Fernández et al., 2001)	61

Figure 2.26	Utility Curve Based on Designer Preferences (Fernández et al., 2001)	62
Figure 2.27	The Deflection of a Cantilever Beam	62
Figure 2.28	Utility Curves for Cost (A) and Maximum Deflection (B)	64
Figure 2.29	Theoretical Validation Established in Chapter 2	68
Figure 3.1	The Role of Chapter 3 in this Thesis	73
Figure 3.2	Discrete Demand Scenario for Cantilever Beam Example	76
Figure 3.3	Linear Demand Scenario for Cantilever Beam Example	78
Figure 3.4	Gaussian Demand Scenario for Cantilever Beam Example	80
Figure 3.5	The Space of Capacity for the Cantilever Beam Example	86
Figure 3.6	Applying PPCTM Comcepts to Production Process Design	89
Figure 3.7	Synthesis of Modes of Managing Process Variety for the Cantilever Beam Example	91
Figure 3.8	Graphical Representation of the Process Platform Design of the Cantilever Beam Example	91
Figure 3.9	The Six Steps of the Product Platform Constructal Theory Method	92
Figure 3.10	Augmented Step 5 of the PPCTM: Infusing the u-cDSP	94
Figure 3.11	Augmenting the Product Platform Constructal Theory Method	99
Figure 3.12	Six Steps of the Augmented PPCTM	99
Figure 3.13	Formulation of the Multi-Stage Utility-Based Compromise Decision Support Problem	106
Figure 3.14	General Solution Method for the Augmented Product Platform Constructal Theory Method	108
Figure 3.15	Theoretical Structural Validation Established in Chapter 3	113
Figure 4.1	The Role of Chapter 4 in this Thesis	116
Figure 4.2	Schematic of Pressure Vessel Embodiment (Hernandez, 2001)	119

Figure 4.3	The Six Steps of the Augmented PPCTM	124
Figure 4.4	Pressure Vessel Space of Customization	125
Figure 4.5	Dimensional Customization of the Length of the Shell (Hernandez, 2001)	127
Figure 4.6	Modular Combination of Vessels (Hernandez, 2001)	127
Figure 4.7	The First Space Element of the Pressure Vessel Space for Customization (Hernandez, 2001)	129
Figure 4.8	The Second Space Element of the Pressure Vessel Space for Customization (Hernandez, 2001)	130
Figure 4.9	The Third Space Element of the Pressure Vessel Example (Hernandez, 2001)	131
Figure 4.10	An Example of the Modular Combination of Vessels (Hernandez, 2001)	131
Figure 4.11	Hierarchic Organization of the Modes for Managing Product Customization for the Pressure Vessel	132
Figure 4.12	Formulation of the Multistage Compromise Decision Support Problem for the Pressure Vessel; First Stage Element	137
Figure 4.13	Formulation of the Multistage Compromise Decision Support Problem for the Pressure Vessel; Second Stage Element	139
Figure 4.14	Formulation of the Multistage Compromise Decision Support Problem for the Pressure Vessel; Third Stage Element	142
Figure 4.15	Sequential Solution Process of the Three Stages	142
Figure 4.16	Solution Algorithm for Pressure Vessel Example	144
Figure 4.17	Exhaustive Search of Design Parameters for the Pressure Vessel Example	150
Figure 4.18	Demand Scenario #1, "Discrete Pyramid Large"	152
Figure 4.19	Demand Scenario #2, "Discrete Pyramid Small"	152
Figure 4.20	Demand Scenario #3, "Linear Large"	153

Figure 4.21	Demand Scenario #4, "Linear Small"	153
Figure 4.22	Demand Scenario #5, "Normal Center Small"	154
Figure 4.23	Demand Scenario #6, "Normal High"	154
Figure 4.24	Demand Scenario #7, "Normal Low"	155
Figure 4.25	Demand Scenario #8, "Normal Center Large"	155
Figure 4.26	Demand Scenario #9, "Random Large"	156
Figure 4.27	Demand Scenario #10, "Random Small"	157
Figure 4.28	Resulting Space of Customization for Demand Scenario #1, "Discrete Pyramid Large"	158
Figure 4.29	Empirical Structural and Performance Validity Established in Chapter 4	170
Figure 5.1	The Role of Chapter 5 in this Thesis	175
Figure 5.2	Hearing Aid Shell (Masters, 2000)	180
Figure 5.3	The Fused Deposition Modeling Process (Williams, et al., 2003)	181
Figure 5.4	Quality Metric Diagram	186
Figure 5.5	The Six Steps of the Augmented PPCTM	190
Figure 5.6	Hearing Aid Shell Space of Capacity	191
Figure 5.7	The First Space Element of the Hearing Aid Space of Capacity	196
Figure 5.8	The Second Space Element of the Hearing Aid Space of Capacity	197
Figure 5.9	The Third Space Element of the Hearing Aid Space of Capacity	198
Figure 5.10	The Fourth Space Element of the Hearing Aid Space of Capacity	199
Figure 5.11	Visualization of the Fourth Stage of the Hearing Aid Example Problem	200
Figure 5.12	Hierarchic Organization of the Modes for Managing Process Customization for the Hearing Aid Example	201

Figure 5.13	Decision Formulation for "Decision 0" of Hearing Aid Example	205
Figure 5.14	Decision Formulation for the First Space Element	208
Figure 5.15	Decision Formulation for the Second Space Element	211
Figure 5.16	Decision Formulation for the Third Space Element	213
Figure 5.17	Decision Formulation for the Fourth Space Element	215
Figure 5.18	Sequential Solution Process of the Four Stages of the Hearing Aid Example	216
Figure 5.19	Solution Algorithm for the Hearing Aid Example	218
Figure 5.20	The Three Major Decisions of the Hearing Aid Example	219
Figure 5.21	Example Space of Capacity for Illustration of Decision 0	220
Figure 5.22	Solution Process of Hearing Aid Example	225
Figure 5.23	ISIGHT Automated Solution Cycle	226
Figure 5.24	Graphical Representation of Resultant Process Platform for Hearing Aid Example	232
Figure 5.25	Convergence History of SQP for One FDM Maxum	234
Figure 5.26	Convergence History of GRG for One FDM Maxum	235
Figure 5.27	Empirical Structural and Performance Validity Established in Chapter 5	245
Figure 6.1	The Role of Chapter 6 in this Thesis	249
Figure 6.2	Validation Strategy Related to Thesis Format	252

LIST OF BOXES

Box 2.1	Defining the Market Space of the Cantilever Beam Tutorial Example	38
Box 2.2	Defining the Objective Function for the Cantilever Beam Tutorial Example	40
Box 2.3	Identifying the Modes for Managing Product Variety for the Cantilever Beam Tutorial Example	41
Box 2.4	Identifying the Number of Stages and Defining a Baseline Decision for Each Stage of the Cantilever Beam Tutorial Experience	43
Box 2.5	Formulating the Multistage Problem for the Cantilever Beam Example	47
Box 2.6	Solving the Dynamic Program of the Cantilever Beam Tutorial Example	50
Box 2.7	The Impact of the Limitations of the PPCTM on the Cantilever Beam Tutorial Example	54
Box 2.8	Assessing Utility Functions for the Cantilever Beam Tutorial Example	62
Box 2.9	Formulating a Multi-Attribute Utility Function for the Cantilever Beam Tutorial Problem	65
Box 2.10	Formulating System Goals for the Cantilever Beam Tutorial Problem	66
Box 2.11	Formulation of the Deviation Function for the Cantilever Beam Example	67
Box 3.1	Modeling Demand of the Cantilever Bean Tutorial Example as a Discrete Function	76
Box 3.2	Modeling Demand of the Cantilever Beam Tutorial Example as a Linear Function	77
Box 3.3	Modeling Demand of the Cantilever Beam Tutorial Example as a Gaussian Distribution	80

Box 3.4	The Process Platform Problem of the Cantilever Beam Tutorial Example	84
Box 3.5	Space of Capacity for the Cantilever Beam Tutorial Example	86
Box 3.6	Modes of Managing Process Variety in the Cantilever Beam Tutorial Problem	88
Box 3.7	Applying PPCTM to the Process Platform Problem of the Cantilever Beam Tutorial Example	90
Box 3.8	Formulation of Objective Functions for the Cantilever Beam Tutorial Problem	93
Box 3.9	Defining a Multi-Attribute u-cDSP for each Decision Stage of the Cantilever Beam Tutorial Problem	95

GLOSSARY

Adjustable Design. Designing products or processes with adjustable features or components to satisfy individual customer's requirements.

Constructal Theory. The concept that hierarchic organizations observed in nature are the result of a sequential optimization process with the objective of the maximization of access (or minimization of resistance, or losses). The optimization process should proceed in a specific time direction: from the optimization of basic elements to the higher-order assemblies of the structure.

Dimensional Customization. Permanent adjustment of a product by a manufacturer to fit a customer's specification.

Economy of Scale. An economy or business that seeks to gain the advantages found in large-scale production. Relies on the notion that unit costs are typically less for large productions than those that are smaller. Success is determined by the number of products made and the speed at which they are produced.

Economy of Scope. An economy or business that seeks to gain the advantages found through sharing production and distribution resources across multiple products. Its purpose is to discover the degree to which different products benefit from shared inputs. Relies on the notion that the integration of products will provide lower production costs.

Hierarchic System. A system composed of interrelated subsystems that are composed in turn of interrelated subsystems until a lowest elementary level of components is reached.

Mass Production. A manufacturing paradigm that strives to lower costs by manufacturing products with little variety on a large scale. Connected to the concept of economies of scale.

Mass Customization. A manufacturing paradigm that strives to offer large amounts of product variety at efficiencies and speeds close to that of mass production. Connected to the concept of economies of scope.

Mode of Managing Product Variety. Any generic approach in product design or its manufacturing process for achieving a product customization

Mode of Managing Process Variety. Any generic approach in the design of a production process for achieving a change in manufacturing capacity.

Modular Design. Arrangement of functions and/or components of a product into modules with standardized interfaces to facilitate the realization of product variants.

Multi-Stage Decision Process. A process that can be separated into a number of sequential stages, which may be completed in one or more ways.

Problem of Optimal Access. The need to determine the optimal bouquet of paths that link all points within an area with a single common destination.

Product Platform. A set of common components, modules, or parts from which a stream of derivative products can be created. Through the development of a product platform, external product variety is inexpensive due to the low amount of internal product variety.

Product Platform Constructal Theory Method (PPCTM). A theory and method for the top-down development of product platforms for customizable products. Based on

the tenets of Constructal Theory and Hierarchic Systems Theory, the PPCTM formulates the design of a product platform as a problem of optimal access in a geometric space.

Production Process Platform. A set of common processes parameters from which a stream of derivative parameters can generate a customized product efficiently for changes in capacity requirement.

Space Element. In constructal theory, a portion or region of the space considered in an access problem, and which is formed by smaller space elements that contain, in turn, smaller space elements until a smallest space element is reached.

Space of Customization. The set of all feasible combinations of values of product specifications that a manufacturing enterprise is willing to satisfy.

Space of Capacity. The range of production capacity that a manufacturing enterprise is willing to satisfy with a production process platform

Utility-based compromise Decision Support Problem (u-cDSP). An extension of the compromise Decision Support Problem (cDSP), the u-cDSP is a decision support construct that is based on utility theory and permits mathematically rigorous modeling of designer preferences such that decisions can be guided by expected utility in the context of risk or uncertainty associated with the outcome of a decision.

Validation Square. A framework for validating design theory and methods in which the 'usefulness' of a design theory or method is associated with whether it provides solutions correctly (structural validity) and whether it provides correct design solutions (performance validity).

NOMENCLATURE

$A_i(X)$	System goals
$AvgProfit$	Average profit of selling family of pressure vessels, $
C	Cost of cantilever beam; $
C	largest distance from the neutral surface of the cantilever beam; mm
C_{batch}	Cost of building a single batch of hearing aid shells; $
$C_{circweld}$	Cost of pressure vessel circular weld, $
C_{equip}	Cost of purchasing of manufacturing equipment for the pressure vessel, $
C_{labor}	Cost of labor to operate FDM machines; $/hr
$C_{longweld}$	Cost of pressure vessel long weld, $
$C_{machine}$	Cost of purchasing FDM machine; $
C_{maint}	Annual cost of maintaining FDM machine; $
$C_{material}$	Cost of FDM material; $/mm3
$C_{material}$	Cost of pressure vessel material, $
C_{order}	Cost of placing an order for the pressure vessel, $
C_p	Cost constant for pressure of pressure vessel, $
C_v	Cost constant for volume of pressure vessel, $
C_{waste}	Cost of pressure vessel waste, $
D	Demand; number of products
D_d	Demand of hearing aid shells per day
d_i^-, d_i^+	deviation variables

E	Modulus of elasticity of cantilever beam; Mpa
f	Objective function
f_{ik}^*	Optimal objective function for stage i; value of the objective function if an optimal policy is followed from stage i to the last stage k
f_i	The value of the objective function at the end of stage i
$g_i(X)$	System constraint functions
h	Part height; mm
I	Moment of inertia of cantilever beam; mm4
k	Scaling constant of u-cDSP
l	Length of hearing aid shell; mm
L	Length of pressure vessel, m
L_o	Raw length of pressure vessel material, m
M^N	A market space with N dimensions
N_b	Number of batches to manufacture hearing aid shells
N_l	Number of layers of hearing aid shell
N_{pb}	Number of parts per batch
N_{ppy}	Number of hearing aids produced in one year
N_{setup}	Number of setups of FDM machine
O	Objective function value
P	Pressure of pressure vessel, MPa
$Profit$	Total profit of selling pressure vessel, $
Q	Quality of hearing aid shell; mm
$\boldsymbol{r}$	A product in the market space

R	Radius of pressure vessel, m
r_i	The ith attribute of a concept
t_{avg}	Average build time of the family of hearing aid shells; sec
t_{base}	Time to build the base of the part; sec
t_{batch}	Total time to build a batch of hearing aid shells; sec
$t_{boundary}$	Time to draw outer and inner boundaries of layer; sec
$t_{cycletime}$	Total time to manufacture a hearing aid shell capacity requirement; sec
t_{fill}	Time to fill the inner portion of the layer via raster scan; sec
T_h	Thickness of pressure vessel head, mm
t_l	Time to build each layer of hearing aid shell; sec
t_{lower}	Time to move down z-axis one layer thickness; sec
t_s	FDM machine setup time; sec
T_s	Thickness of pressure vessel shell, mm
t_{supp}	Time to draw the support material for this layer; sec
$U(X)$	Overall, multi-attribute utility function
$u_i(A_i(X))$	Utility function for each goal
V	Volume of pressure vessel, m3
$V_{material}$	Volume of material used to build one hearing aid shell; mm3
V_{ss}	Scan speed of FDM machine; mm/s
w	Displacement of cantilever beam; m
w_l	Width of lower ellipse diameter
w_{road}	Road width of FDM deposit; mm
w_u	Width of upper ellipse diameter; mm

Z	Devation function
ΔD	Range of capacity in the hearing aid shell space of capacity
ΔD_i	The dimension of the ith space element in the direction C of the market space; a decision variable for a stage i.
ΔL	Range of cantilever beam lengths in the market space
ΔP	Range of cantilever beam end loads in the market space
ΔP	Range of pressures in the pressure vessel market space
ΔP_i	The dimension of the ith space element in the direction P of the market space; a decision variable for a stage i.
Δr	a derived product realization in the market space
ΔV	Range of volumes in the pressure vessel market space
ΔV_i	The dimension of the ith space element in the direction V of the market space; a decision variable for a stage i.
μ	Mean of demand (Gaussian distribution)
ρ	Density of welding material, kg/m3
σ	Standard deviation of demand (Gaussian distribution)
σ_y	Yield strength of pressure vessel material, Mpa

SUMMARY

Offering customers variety in products has given numerous companies a valuable competitive advantage. This transition from mass production to mass customization has increased the need for shortened product development time. As a result, designers are faced with the need for efficient and effective methodologies for designing customized products.

The Product Platform Constructal Theory Method (PPCTM) was developed by Dr. Gabriel Hernandez to facilitate the synthesis of multiple modes for managing product variety in the development of product platforms for customized products. Through the application of the PPCTM, the design of platforms for customizable products is represented as a problem of access in a geometric space. The result of the use of the PPCTM is a hierarchical organization of the modes for managing product variety, as well as the specification of their range of application across the product platform.

The focus in this thesis is the augmentation of this product platform design method in order to alleviate two of its major limitations: an inability to cope with multiple objectives, and a reliance on the assumption of uniform demand across the market space. Specifically, three components are infused into the formulation of the PPCTM in this thesis. Several different demand-modeling techniques are presented in order to give a designer the power to capture and convey multiple types of demand information for the market being analyzed. The utility-based compromise Decision Support Problem (u-cDSP) is infused into the multistage decision formulation stage in order to empower a

designer with the ability to make decisions in the presence of multiple objectives. Finally, several new concepts are introduced in order to apply the augmented PPCTM to the development of the manufacturing process of a customized product.

The resulting augmented PPCTM provides a designer with the ability to synthesize multiple modes for managing variety in the development of both product and process platforms for multiple objectives and for markets of non-uniform demand. This augmented product platform for the development of a line of customizable pressure vessels, and the design of a production process platform for the manufacture of customizable hearing aid shells.

CHAPTER I

THE CHALLENGES OF DESIGNING FOR MASS CUSTOMIZATION

In this thesis the Product Platform Constructal Theory Method (PPCTM) is augmented in order to provide a designer with the ability to synthesize multiple modes for managing customization in the development of both product and process platforms for multiple objectives and for markets of non-uniform demand. Specifically, these augmentations include:

1) Several different demand-modeling techniques are presented in order to give a designer the power to capture and convey multiple types of demand information for the market being analyzed. Also, the formulation of objective functions is generalized in order to cope with these different types of demand scenarios.
1) The utility-based compromise Decision Support Problem (u-cDSP) is infused into the multistage decision formulation stage in order to empower a designer with the ability to make decisions in the presence of multiple objectives.
2) Finally, the underlying theory of the PPCTM is abstracted in order to apply the its augmented form for the design of a production process.

The Product Platform Constructal Theory Method was developed by Gabriel Hernandez in order to provide designers a methodical approach for synthesizing multiple modes of managing process customization in the development of product platforms for customized products (Hernandez, 2001). As a result of the PPCTM's theoretical foundations in both hierarchical systems theory and constructal theory, the design of

platforms for customizable products is represented as a problem of access in a geometric space. The result of the use of the PPCTM is a hierarchical organization of the modes of managing customization, as well as the specification of their range of application across the product platform.

The focus in this thesis is the augmentation of this product platform design method in order to alleviate two of its major limitations: an inability to cope with multiple objectives, and a reliance on the assumption of uniform demand across the market space. The resulting augmentations, and application to the realm of product process design, greatly add to the value of the PPCTM as a systematic method in which to design and manufacture products for mass customization.

1.1 BACKGROUND AND MOTIVATION FOR STUDYING MASS CUSTOMIZATION

The Product Platform Constructal Theory Method is to be used primarily for the design of product platforms for mass customized products. In order to establish a context for this research, a description of mass customization is presented in this section. Specifically, reasons for, and the challenges associated with, pursuing mass customization is discussed in Sections 1.1.1 and 1.1.2 respectively.

1.1.1 The Contemporary Competitive Advantage: Product Variety

Henry Ford's attitude expressed in the phrase, "You can have any color car as long as it's black," no longer satisfies consumers. This has been evident as early as 1924, when General Motor's Alfred Sloan responded with "A car for every purse and purpose" (July 2003). Companies are discovering that customers cannot be lumped into a homogenous group. Many of today's customers seek unique purchases, high quality, and

an enterprise that can meet their specific product needs. As a result, many manufacturing enterprises are increasingly accommodating the individual needs of customers by providing product variety or customization in their products. Evidence of this trend can be observed in Figure 1.1 below.

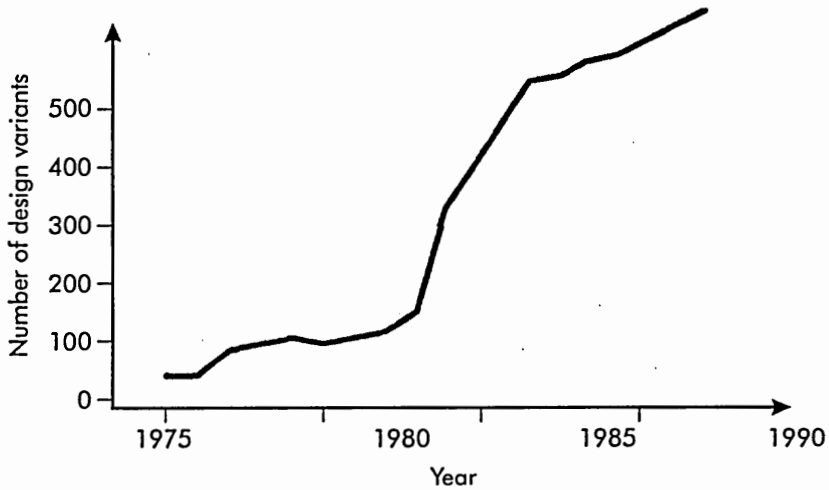
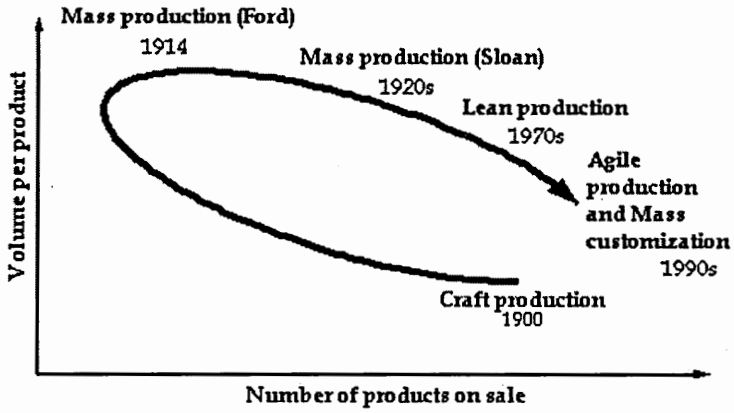


Figure 1.1 – Number of Car Variants at BMW from 1975 to the Early 1990s (Ericsson et al., 1999)

In terms of competitiveness, it is more advantageous for a manufacturing enterprise to create (and offer) change instead of reacting to it. **Mass customization** has emerged as a manufacturing paradigm for enterprises to efficiently and effectively satisfy customer's requirements for variety. As shown in Figure 1.2, just as the face of manufacturing in the United States once shifted away from craft guilds to the *economies of scale* of mass production, today's manufacturing style is shifting towards the *economies of scope* of mass customization. At its best, mass customization is the developing any product for anybody, anywhere, anytime, and at any volume; it is the marriage of the craft guilds' unique products and the large volumes of mass production.



Characteristic	*Craft Production*	*Mass Production*	*Lean Production*	*Mass Customization*
Anybody	No	Yes	Yes	Yes
Anything (any product, any design)	Yes	No	No	Yes
Anywhere	No	Yes	Yes	Yes
Any time	No	Yes	Yes	Yes
Any Volume	No	No	Yes	Yes

Figure 1.2 – Anybody, Anything, Anywhere, Anytime, Any Volume: The Five Goals of Customized Manufacturing (myb20.com, 2001)

Mass customization is an oxymoron. It involves producing a product for an individual, but distributing it on a mass scale. This is accomplished by satisfying numerous customers with demands for variety through products that, from a product development standpoint, have much in common. Another key attribute of mass customization is that it involves the production of customized products at efficiencies and speeds close to those of mass production (Anderson, 1997). Pine (Pine, 1993) summarizes mass customization with:

> At its core (mass customization) is a tremendous increase in variety and customization without a corresponding increase in costs. At its limit, it's the mass production of individually customized goods and services. At its best, it provides strategic advantage and economic value.

Mass customized production involves building a product in response to a particular customer's desire, as opposed to producing a generic product to be placed in an inventory

in hopes that some customer will later purchase it. As can be seen in Figure 1.3, although mass production has an economic advantage in high volume production, the ability to satisfy individual customer needs can translate into higher profit margins.

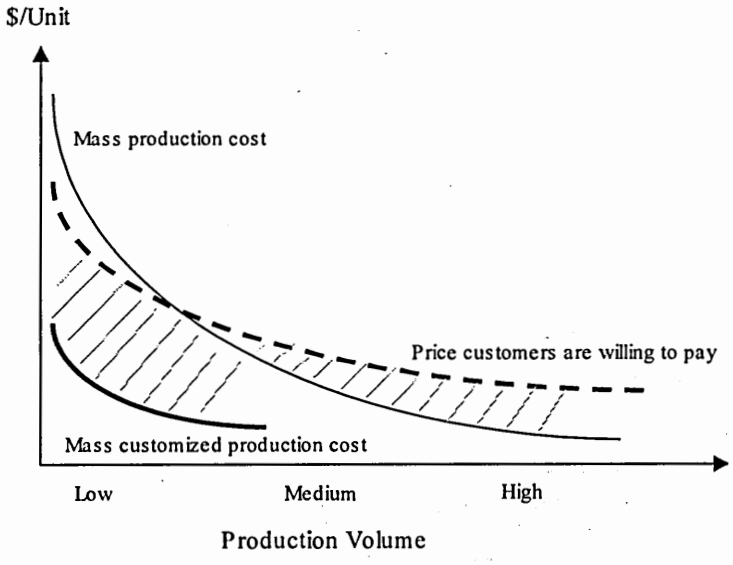


Figure 1.3 – The Economic Implications of Mass Customization (Tseng et al., 1998)

This is the key advantage of pursuing mass customization. For the appropriate product, the application of mass customization techniques enables manufacturing enterprises to satisfy their customers' demand for variety cost effectively.

This advantage leaves many questions unanswered however. Questions like, "What are some 'techniques' of mass customization?" "How should a product be designed to be able to foster the mass customization paradigm?" "How can a group of products, with inherent variety, be manufactured in a cost effective manner?" More questions and further challenges of implementing mass customization are provided in the next section.

1.1.2 The Challenges of Mass Customization

The challenges involved with designing a product and its manufacturing process for customization are inherent in the differences between mass production and mass customization. These differences are detailed in Table 1.1.

Table 1.1 – The Differences Between Mass Production and Mass Customization (Pine, 1993)

	Mass Production	*Mass Customization*
Focus	Efficiency through stability and control	Variety and customization through flexibility and quick responsiveness
Goal	Developing, producing, marketing, and delivering goods and services at prices low enough that nearly everyone can afford them	Developing, producing, marketing, and delivering affordable goods and services with enough variety and customization that nearly everyone finds exactly what they want
Key Features	Stable demand	Fragmented demand
	Large, homogeneous markets	Heterogeneous niches
	Low-cost, consistent quality, standardized goods, and services	Low-cost, high-quality, customized goods and services
	Long product development cycles	Short product development cycles
	Long product life cycles	Short product life cycles

The key differences outlined in the table above – mass customization's fragmented demand, heterogeneous niches, and short product development and life cycles – are the also the key challenges in the design of a customized part.

The crux of mass customization, the offering of variety, is the foremost difficulty that enterprises are faced with in making the transition. How can one affordably offer a customized product? Anderson addresses this question through the identification of the cost of variety (Anderson, 1997). The cost of variety is the sum of all the costs of attempting to offer customers variety with inflexible products that are produced in inflexible factories and sold through inflexible channels. This cost includes:

- Cost of customizing or configuring products
- Cost of excessive parts, and parts of excessive variety
- Cost of excessive procedures
- Cost of excessive processes and operations
- Cost due to slow responsiveness to customer needs or changing market conditions
- Cost of insufficient product development
- Cost of lost sales due to stock-outs

The key to mass customization, therefore, is the development of a product and a production process that minimizes the components of this cost.

As shown in Figure 1.4, a manufacturing system focused on mass customization is able to keep the cost of variety at a minimum when market variety increases when compared to mass production.

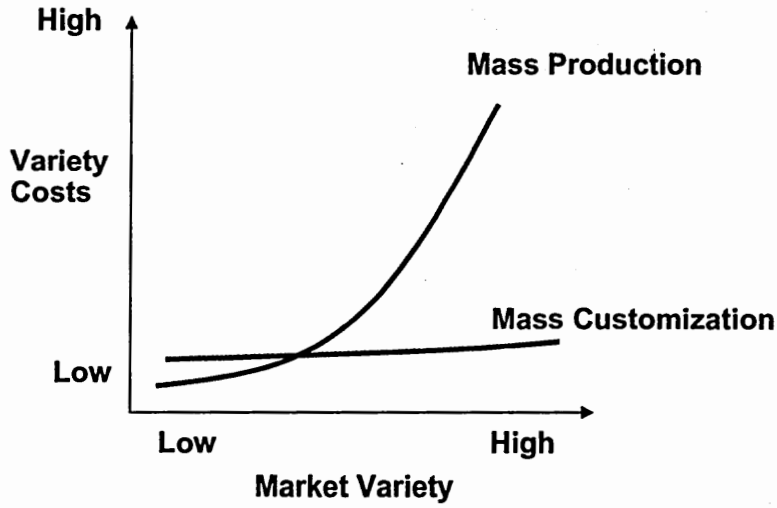


Figure 1.4 – The Cost of Variety as a Function of Market Variety (Anderson, 1997)

When faced with the need for product variety, the traditional mass production reaction is to increase inventory to cope with the new demand. It is impossible, however, to cost effectively cope with customers' demands for product variety through an increase in inventory. If products are not designed appropriately, then coping with product variety and customization can be a slow and costly ordeal. Hence, manufacturing enterprises are recognizing that product design presents the best control over offering such a variety (Anderson, 1997). The core issue of transitioning to mass customization now becomes how one can design a product for affordable customization.

It is noted that, for mass customization, the design of the product must be concurrent with the design of its production process. A strong product design cannot be implemented without an equally strong production strategy. While the emergence of

flexible manufacturing technologies like CNC machining, laser cutting, parametric CAD/CAM packages, assembly robotics, and rapid prototyping have provided enterprises the tools for producing a variety of products, the production process itself must be designed to be able to handle the fragmented demand of mass customization (see Table 1.1).

Product and manufacturing process design has been identified as the key to successful implementation of mass customization. As a result, an exploration of available methods for the design of mass customized products is conducted in Section 1.2.

1.2 THE NEED FOR SYSTEMATIC PRODUCT PLATFORM DESIGN METHODS

Typically, mass customization is accomplished through developing product platforms – a set of common components, modules or parts from which a stream of derivative products can be created (Lehnerd, 1987). The reasoning behind platform design is to (1) simplify the product offering and reduce part variety by (2) standardizing components so as to (3) reduce manufacturing and inventory costs and (4) reduce manufacturing variability (i.e., the variety of parts that are produced in a given manufacturing facility) and thereby (5) improve quality and customer satisfaction (Simpson, 1999). The development of product platforms enables companies to offer external product variety and customization to the customer with minimum internal component variety.

Although the use of product platform development offers an alternative to designing customized products, it alone is not the complete solution. Referring back to Table 1.1,

the average product life cycle has significantly shortened within the past decade. This can be observed in Figure 1.5, where the life cycles of three Philips brand electronic media products (VHS, DVD, and DVD-R) are compared.

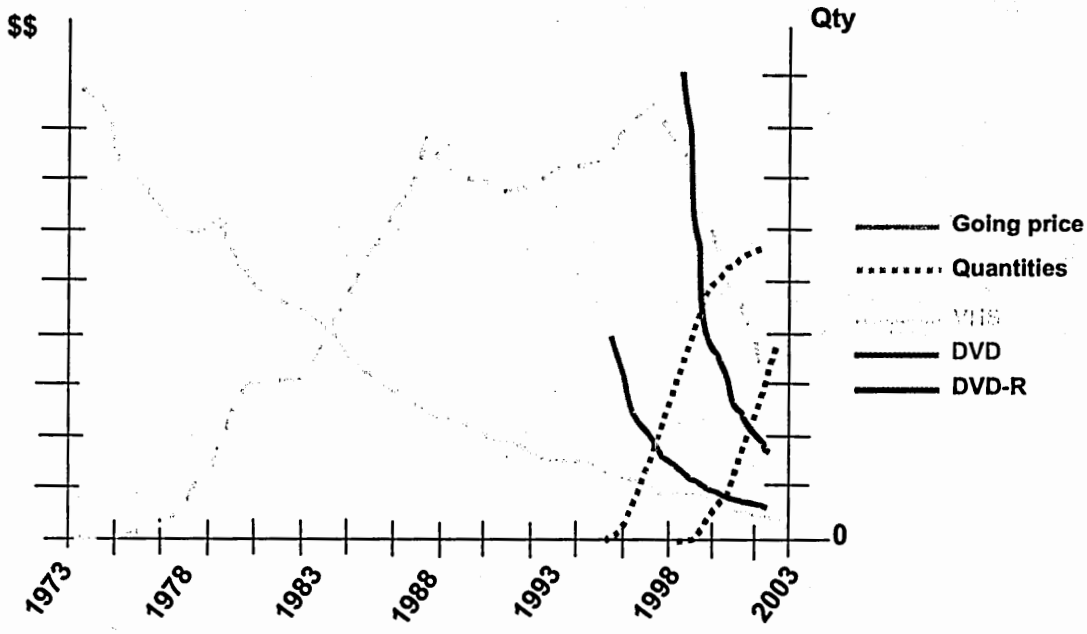


Figure 1.5 – Average Life Cycle of Philips Electronic Media (Minderhoud, 2003)

Momentarily ignoring the labels of the y-axes, one can compare the average life cycles of these three products by simply observing the length of their relative curves. As can be seen, the more frequently reduced products, the DVD and the DVD-R, have experienced much shorter life cycles.

As product life cycles are shortened, the product development life cycle becomes even more compressed (refer to Table 1.1). As such, the need for a *systematic* product platform development method is needed. Tseng has corroborated this need: "it is imperative to develop a coherent framework within which systematic approaches can be taken to enable the realization of mass customization," (Tseng et al., 1998). Developing systematic approaches for product platform design is therefore a pressing need and a relevant research problem.

Various engineering approaches have been proposed to develop product platforms and can be classified as either *bottom-up* or *top-down* (Simpson, 1999). Bottom-up approaches are the redesign and consolidation of existing products to create more competitive product families by reducing part variety and standardizing components. Top-down platform design is characterized by an up-front decision to develop simultaneously a product family based on a common core and reduce redesign cost.

1.2.1 Bottom-up Approaches

In general, the main advantage of bottom-up approaches is the existing knowledge of parts and processes. Their main disadvantage is that product rationalization is achieved after a number of products have been designed and manufactured.

Group Technology

The crux of *Group Technology* (Kalpakjian, 1997) is design-oriented grouping. Under group technology, part families are formed with members that use similar processes based on domain independent classification and coding schemes, where existing parts are grouped in order to prevent design proliferation and redundancy.

Modular Function Deployment

Ericsson and Erixon's *Modular Function Deployment* (Ericsson et al., 1999) creates product families through modular design. By identifying "module drivers" the designer uses QFD tools to develop and select technical solutions to the design problem.

Product Reasoning System

Siddique and Rosen (Siddique et al., 2000) propose a *Product Family Reasoning System* for replacing an existing product family with an improved one. Specifically, they

focus on the components in a product family, and their arrangements and relationships through configuration reasoning methods.

1.2.2 Top-down Approaches

The Product Platform Concept Exploration Method

The *Product Platform Concept Exploration Method* of Simpson and coauthors is intended to facilitate the synthesis and exploration of a robust product platform that can be scaled into a family of products (Simpson et al., 2001). Input to the *Product Platform Concept Exploration Method* is a set of overall product family design requirements; the output is a portfolio of products based upon a common product platform.

Variation-Based Platform Design Methodology

Nayak and coauthors follow a similar approach with their *Variation-Based Platform Design Methodology*, wherein commonalization is performed only a set of carefully chosen (robust) variables; thus somewhat reducing the effect of commonalization of product performance (Nayak et al., 1999).

1.2.3 Limitations of Existing Top-Down Methods

There are two perceived limitations of top-down approaches – the first is related to the "extent" of commonality, the second, to the number of varied specifications. Regarding the first limitation, top-down approaches have been used to commonalize features or components across an entire product family (that is, the feature is made common to all the relevant product variants). Such a commonalization can result in a significant loss of performance, and has been discussed in the literature (e.g., (Nayak et al., 1999; Simpson et al., 1999; Seepersad et al., 2000; Nelson et al., 2001).

In an effective platform strategy, one should be able to specify different levels of commonality for the various features and components of the product family in order to reduce the impact of commonality on performance. A method for determining platform extent that is based on the compromise Decision Support Problem (coupled with concepts derived from linear physical programming) has been previously proposed (Seepersad et al., 2000). Unfortunately, this approach can only be utilized with a limited number of products. Designing products for mass customization implies a very large number of product variants for which this approach is impractical.

Regarding the second limitation, existing top-down approaches have typically considered variety only in one product specification (e.g., the torque of a motor). Typically, products are customized for multiple specifications (e.g., the torque and the power of the motor) using multiple approaches for managing product variety (e.g., modular design, adjustable features, dimensional customization with CAD/CAM technology, etc.).

As a result of these limitations, designers using the existing top-down approaches are unable to systematically design the complex product platforms that are typical of those realized in industry. An example of the combining of multiple approaches of commonality is Matsushita's customized bicycles (Anderson, 1997): modularity is applied in wheels, pedals, drive chain, seats, handlebars and controls; adjustable customization is used to allow adjustments for the height and angle of the seat and handlebars; and dimensional customization enables customized lengths of the frame tubing.

1.2.4 Product Platform Design as a Problem of Access in a Geometric Space

In lieu of the two limitations of top-down approaches listed in the previous section, Hernandez proposes a novel top-down approach for developing product platforms that facilitates the realization of a stream of customized product variants, and which accommodates the issue of multiple levels of commonality and multiple customizable specifications (Hernandez, 2001; Hernandez et al., 2002; Hernandez et al., 2003). The result of this research, accomplished for his doctoral dissertation, is the *Product Platform Constructal Theory Method (PPCTM)*.

Specifically, Hernandez shows that it is feasible and useful to formulate and solve systematically the design of hierarchic product platforms for customizable products as a problem of optimization of access in a geometric space (Hernandez et al., 2002). The result of the use of the PPCTM is a hierarchical organization of several multiple approaches of commonality, as well as the specification of their range of application across the product platform. A detailed description of Hernandez's PPCTM is provided in Section 2.1.

As mentioned in the introduction of this chapter, the PPCTM has its own set of limitations. The augmentation of the PPCTM to alleviate these limitations is the primary focus in this thesis. The presentation of the research questions guiding this augmentation process is presented in Section 1.3.

1.3 RESEARCH FOCUS, QUESTIONS, AND HYPOTHESES

There are three substantial limitations of the Product Platform Constructal Theory Method:

2) the working assumption of uniform demand across the market space,

3) the inability for a designer to model multiple objectives,

4) and its negligence of the process design side of mass customization.

Referring once again to Table 1.1, one can observe that mass customization is identified through the fragmented nature of the demand. As such, the alleviation of the first limitation listed is considered to be the main focus in this thesis. The second limitation listed, the inability of a designer using the PPCTM to model multiple objectives, is a direct result of the desire to alleviate the first limitation – the success of the modeling non-uniform demand is dependent of a designer being given the opportunity to make decisions involving multiple objectives. The third limitation is a result of recognizing the need to concurrently design both the product and its production process in implementing mass customization.

As a direct result of these limitations, there are two major thrusts in this work:

1) The *augmentation* of the PPCTM:

   a. the infusion of a decision construct capable of empowering a designer to make decisions with the presence of multiple objectives

   b. the infusion of modeling techniques capable of capturing a wide variety of market demand information

   c. the infusion of concepts capable of expanding the PPCTM's applicability to the design of a manufacturing process which produces customized parts

2) The *validation* and *verification* of the augmented PPCTM through its application to two example problems:

a. the application of the PPCTM to the realm of product design for the development of a line of customizable pressure vessels

b. the application of the PPCTM to the realm of production process design in order for the development of a family of process parameters for the manufacturing of customizable hearing aid shells.

The infusion of the proposed augmentations is illustrated in Figure 1.6.

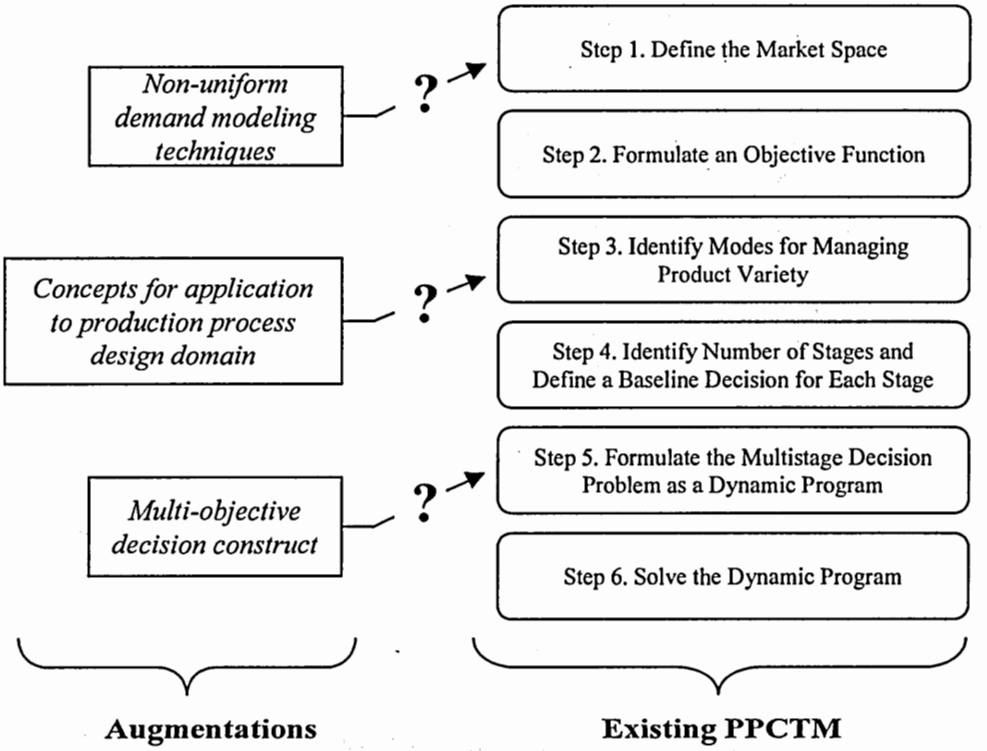


Figure 1.6 – Representation of the Augmentations Proposed in this Thesis

The outcome of this work is the development of a method from which a designer has the ability to develop both a product and a *production process* platform for customized products in markets of non-uniform demand.

The focus in this thesis is further conveyed through a set of research questions and hypotheses. The primary research question to be investigated in this thesis is:

Primary Research Question:
How can the Product Platform Constructal Theory Method be augmented to synthesize multiple modes for managing variet in order to create product and process platforms in a market of non-uniform demand?

This primary research question are addressed by investigating a set of research questions (Q1, Q2, and Q3) and associated hypotheses (H1, H2, and H3).

Q1: *How can the Product Platform Constructal Theory Method be augmented in order to cope with the development of product platforms in a market of non-uniform demand?*

H1: Modeling the demand as a property of the geometric market space, a designer can successfully solve the non-uniform optimal access problem through the discretization of the space.

Q2: *How can the Product Platform Constructal Theory Method be augmented in order to enable a designer to cope with the presence of multiple objectives?*

H2: The infusion of the utility-based compromise Decision Support Problem will provide a rigorous construct in which a designer can make decisions with multiple, conflicted, and coupled objectives.

Q3: *How can the Product Platform Constructal Theory Method be successfully applied to the domain of production process development for customized products in a market of non-uniform demand?*

H3: The generalized application of the PPCTM (and the augmentations proposed in Hypotheses 1 and 2) can be applied to assist a designer with the development of a "process platform" for the manufacture of customized products.

Each of the research questions are addressed in specific sections of the thesis as outlined in Table 1.2.

Table 1.2 – Relevance of Thesis Chapters and Sections to Research Questions and Hypotheses

Relevant Sections:	Chapter 2		Chapter 3				Chapter 4			Chapter 5		
Research Questions	2.1	2.2	3.1	3.2	3.3	3.4	4.1	4.2	4.3	5.1	5.2	5.3
Q1	X		X		X	X	X	X	X	X		
Q2	X	X			X	X				X	X	X
Q3	X			X	X	X				X	X	X

1.4 VALIDATION AND VERIFICATION STRATEGY FOR THIS THESIS

The validation and verification strategy for this thesis is based on the validation square proposed by Pedersen and coauthors (Pederson et al., 2000). In the views of these authors the validation of work in engineering design cannot be pursued in formal, rigorous, quantitative verification based on logical induction and/or deduction because it relies heavily on both subjective statements. Thus Pederson and coauthors propose an alternative approach where knowledge validation in design becomes "a process of building confidence in its usefulness with respect to a purpose."

Pederson and coauthors propose a framework for validating design theory and methods in which the 'usefulness' of the theory or method is associated with whether they provide design solutions correctly (structural validity) and whether it provides correct design solutions (performance validity). This process of validation is presented in the validation square shown in Figure 1.7.

This validation strategy is instantiated in each chapter of this thesis. The specific manner in which the validation square is applied to this thesis is presented in Figure 1.8. Inspired by the MS thesis of Seepersad (Seepersad, 2001), the figure is arranged according to the quadrants of the validation square, and references are included for chapters and sections in which the validation is documented.

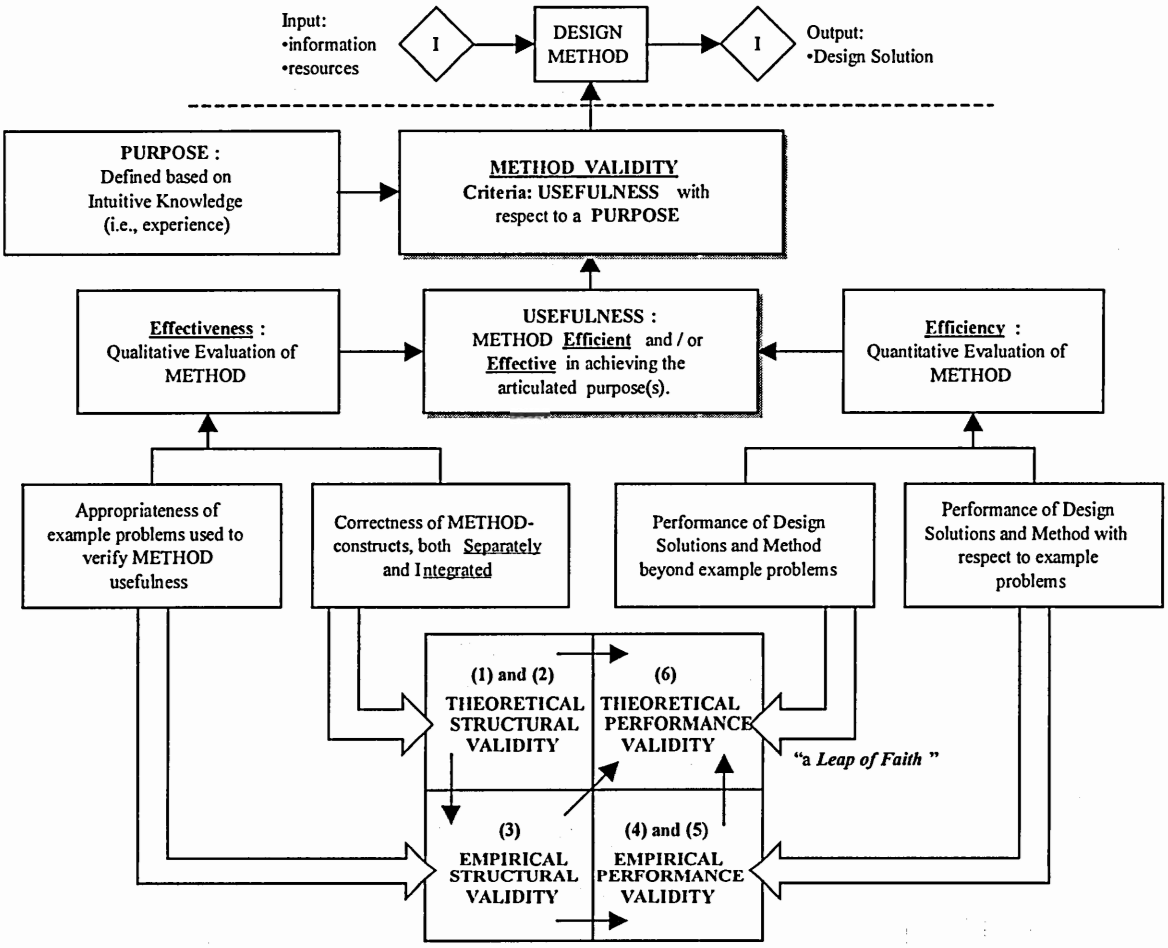


Figure 1.7 – The Validation Square: Validating Design Theories or Methods (Pederson et al., 2000)

Theoretical Structural Validation

Review of product platform design techniques, thorough description of Hernandez's PPCTM (which serves as a foundation for this work), and a discussion of the constructs to be included to deal with non-uniform demand and the presence of multiple objectives.	Chapter 2
Presentation of step-by-step method and an implementation flow-chart for this approach. Explicitly stating the augmentations to be added the PPCTM.	Chapter 3

Empirical Structural Validation

Discuss the appropriateness of the pressure vessel example problem: - Document that the abstracted ideas of the constructs presented are relevant to the application of the pressure vessel example problem. - Document that the pressure vessel example problem is representative of an actual problem for which the approach is intended. - Document that the data associated with the pressure vessel example problem can support a conclusion or conclusions for Hypothesis 1.	Chapter 3, Chapter 4
Discuss the appropriateness of the hearing aid shell example problem: - Document that the abstracted ideas of the constructs presented are relevant to the application of the hearing aid shell example problem. - Document that the hearing aid shell problem is representative of an actual problem for which the approach is intended. - Document that the data associated with the hearing aid shell example problem can support a conclusion or conclusions for Hypotheses 2 and 3.	Chapter 3, Chapter 5

Empirical Performance Validation

Build confidence in the usefulness of the method using the pressure vessel and hearing aid shell example problems.	Chapters 4 and 5

Theoretical Performance Validation

Build confidence in the generality of the approach or in the usefulness of the approach beyond the specific example problems.	Chapter 6

Figure 1.8 – Validation Strategy for this Thesis

1.4.1 Theoretical Structural Validity

With respect to the square, theoretical structural validity involves accepting the individual constructs and premises constituting the theory or method as well as the internal consistency of their assembly into a theory/method. Pederson and coauthors propose that establishing this portion of the square involves searching and referencing the literature related to each of the constructs employed in the design theory or method.

In this thesis theoretical structural validity is presented in Chapters 2 and 3. Critically reviewing each proposed augmentation, and then analyzing the consistency of their combination and infusion into the PPCTM completes this quadrant of the validation square.

1.4.2 Empirical Structural Validity

Empirical structural validity includes building confidence in the appropriateness of the example problems chose for illustrating and verifying the performance of the design method. This consists of documenting that the example problems chosen for validation are similar to the problems for which the methods/constructs are generally accepted, i.e., that the example problems represent actual problems for which the method is intended, and that the data associated with the example problems can be used to support a conclusion.

Empirical structural validity is established in this thesis in Chapters 4 and 5. In these chapters two example problems are developed: the product platform design of a line of customizable pressure vessels, and the process platform design of a production process for a line of customizable hearing aid shells.

1.4.3 Empirical Performance Validity

Empirical performance validity includes building confidence in the usefulness of a method using example problems. Metrics for usefulness should be related to the degree to which the method's purpose has been achieved (e.g., reduced cost, reduced time, improved quality). It is also important to establish that the resulting usefulness is, in fact, a result of applying the theory/method.

As can be seen in Figure 1.8, empirical performance validity is also explored in Chapters 4 and 5. The usefulness of the augmentation is evaluated through the comparison of the results obtained from the use of the augmented PPCTM to results obtained without it.

1.4.4 Theoretical Performance Validity

Theoretical performance validity involves building confidence in the generality of the theory or method and accepting them as useful beyond the example problems. This involves showing that the problems are representative of a general class of problems and that the theory or method is useful for these problems; from this the general usefulness of the theory or method can be inferred. The "leap of faith" that is theoretical performance validity is explored in the final chapter of this thesis.

1.5 ORGANIZATION OF THIS THESIS

An outline of the organization of this thesis is provided in Figure 1.9. When observed in combination with Figure 1.8, the emphasis of the validation of the research hypotheses of this thesis can be seen.

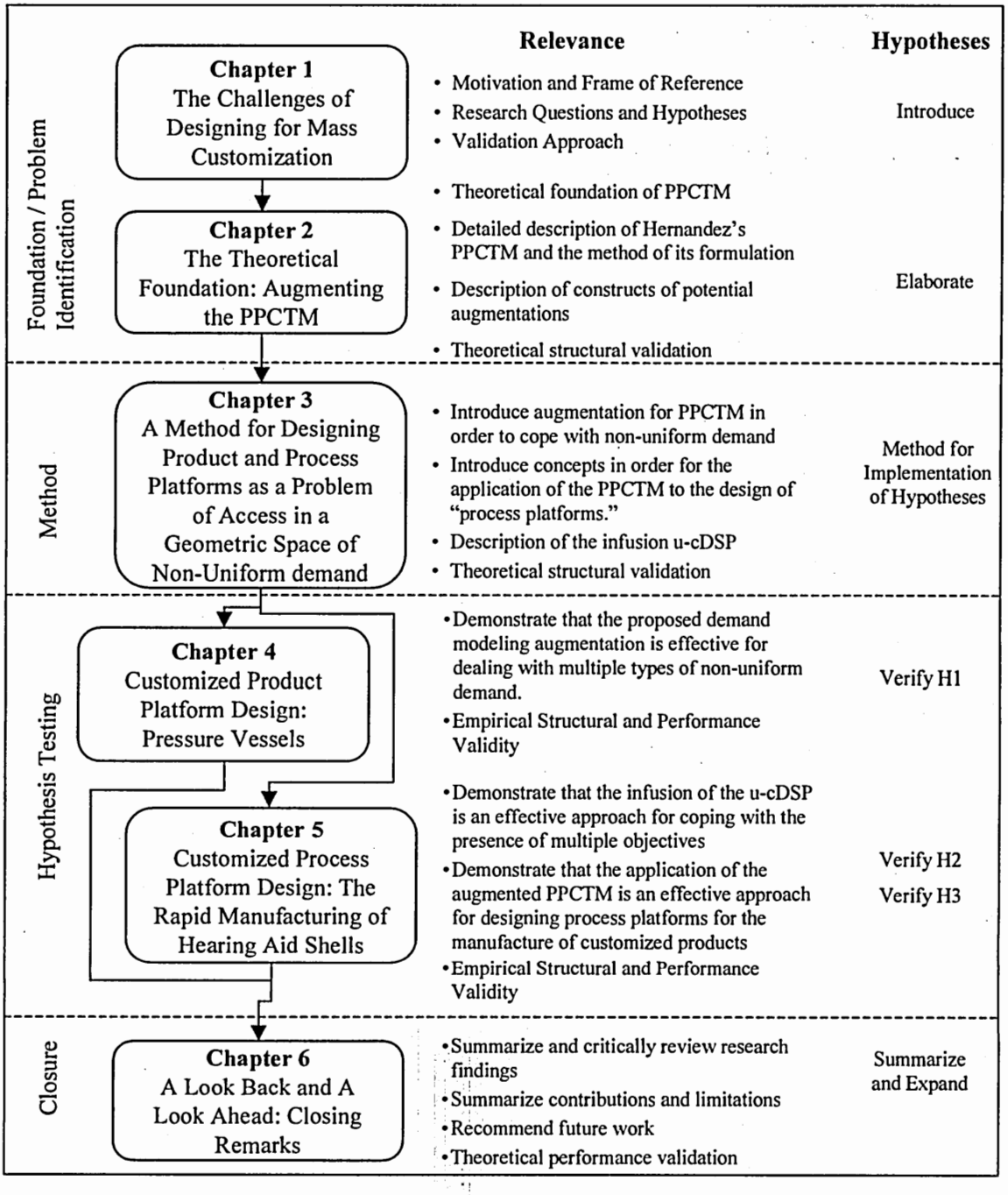


Figure 1.9 – A Roadmap for this Thesis

Theoretical structural validation is addressed in Chapter 2 through the presentation of the theoretical foundation of the Product Platform Constructal Theory Method as well as of its proposed augmentations. In this chapter, the PPCTM is described in detail to provide a foundation on which the augmentation methods are added. The descriptions of the constructs of the utility-based compromise Decision Support Problem (u-cDSP), the augmentation which will provide a decision construct in which multiple objectives can be modeled, is also presented in this chapter.

In Chapter 3, the infusion of the augmentations of the PPCTM is discussed in detail. Emphasis is placed on verifying the internal consistency of the method as a combination of the constructs presented in Chapter 2. Multiple mathematical techniques capable of modeling non-uniform demand are presented at the beginning of this chapter. Concepts, and accompanying definitions, are then presented in order to establish a framework for the successful application of the PPCTM to the domain of process design. Specific details of the infusion of the u-cDSP into the PPCTM are also presented in this chapter. Finally, the resulting six-step augmented PPCTM is presented.

Empirical structural and performance validity is offered in Chapters 4 and 5 through the presentation of two example problems. In Chapter 4 an example of the development of product platforms for customizable pressure vessels is presented. In this example problem, four approaches for managing product variety are synthesized to develop a product platform for customizable pressure vessels for a variety of non-uniform demand scenarios. In Section 4.1 the empirical structural validity is established by documenting that the example is relevant and that the results can support conclusions. In the remainder of the chapter, the example is developed and exercised in order to establish empirical

performance validity. The customizable pressure vessel example is relevant because it involves:

- A classic mechanical engineering example: the selection of design parameters for a pressure vessel
- The synthesis of four modes of managing product variety: dimensional customization of the vessel length, standardization of the plates (the raw material), standardization or radii, and modular combination of vessels to offer variety to customers in both volume and pressure.
- The determination of product platform configurations for multiple non-uniform demand scenarios. Modeled as a linear function of volume and pressure, as a two-dimensional Gaussian (normal) distribution, as discrete changes, and as a random distribution.

Specific objectives for this example problem are:

- Demonstrate that the proposed augmentations to the PPCTM enable its use to cope with a variety of non-uniform demand scenarios (Hypothesis 1).
- Identify the lessons learned from the configurations of product platforms when confronted with multiple demand scenarios.

In Chapter 5, an example of the application of the PPCTM in the domain of production process design is presented. In this example problem, five approaches for managing process customization are synthesized in order to develop a process platform for the rapid manufacturing of customizable hearing aid shells for a wide range of capacity requirements. In Section 5.1 the empirical structural validity is established by documenting that the example is relevant and that the results can support conclusions. In

the remainder of the chapter, the example is developed and exercised in order to establish empirical performance validity. The customizable hearing aid shell example is relevant because it involves:

- A view of a current industry problem: the use of rapid manufacturing to develop customized hearing aid shells.
- The establishment of the concept of a *process platform* – a family of process parameters that can satisfy a variety of demand requirements.
- The synthesis of five modes of managing product variety: altering process parameters (layer thickness, road width and batch size), standardization of machine type, and scaling the machine quantity in order to offer the company variety of capacity.
- The determination of process platform configurations for a wide range of capacity requirement.

Specific objectives for this example problem are:

- Demonstrate that the proposed infusion of the utility-based compromise Decision Support Problem will provide a designer using the augmented PPCTM the ability to cope with multiple objectives.
- Demonstrate that the proposed application of the PPCTM can be used to efficiently and effectively develop a process family for a product of non-uniform demand (Hypothesis 3).
- Identify the lessons learned from the development of a process platform.

Finally, feasibility and usefulness beyond the example problems (theoretical performance validity) of the proposed augmentations is discussed in Chapter 6. The

thesis is summarized and critically reviewed and relevant contributions and avenues for future work are discussed.

The roadmap shown in Figure 1.9 will be presented and updated at the beginning of each chapter of this thesis in order to provide the reader an update of the progress being made towards the validation of the three research hypotheses. Another tool implemented in this thesis is the use of a running icon, shown in Figure 1.10.

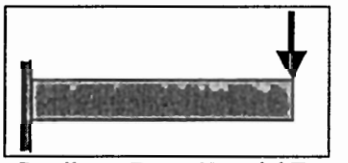


Figure 1.10 – The Cantilever Beam Tutorial Example Running Icon

This icon symbolizes the author's use of a tutorial example involving the design of a line of customizable cantilever beams. This simplified example will be used throughout the thesis to explain difficult concepts, and will be presented in a distinctly defined Box. The concerted use of these organizational tools will hopefully assist in the reader's understanding of this research.

CHAPTER II

THE THEORETICAL FOUNDATION: AUGMENTING HERNANDEZ'S PRODUCT PLATFORM DESIGN METHOD

In this chapter the theoretical foundations for designing platforms as a problem of access in a geometric space with non-uniform demand are investigated. The purpose of this chapter is to present both the Product Platform Constructal Theory Method as well as the fundamentals of the augmentations necessary for it to be capable of answering the research questions presented in Section 1.3. As shown in Figure 2.1 (a selected portion of Figure 1.9), theoretical structural validation is begun in this chapter by reviewing, referencing, and discussing the literature relevant to each of the constructs employed in this work. These constructs include

- the Product Platform Constructal Theory Method (Section 2.1),
- and the utility-based compromise Decision Support Problem (Section 2.2).

These constructs are combined in Chapter 3 in order to present a method for designing product and process platforms in markets of non-uniform demand in lieu of multiple design objectives.

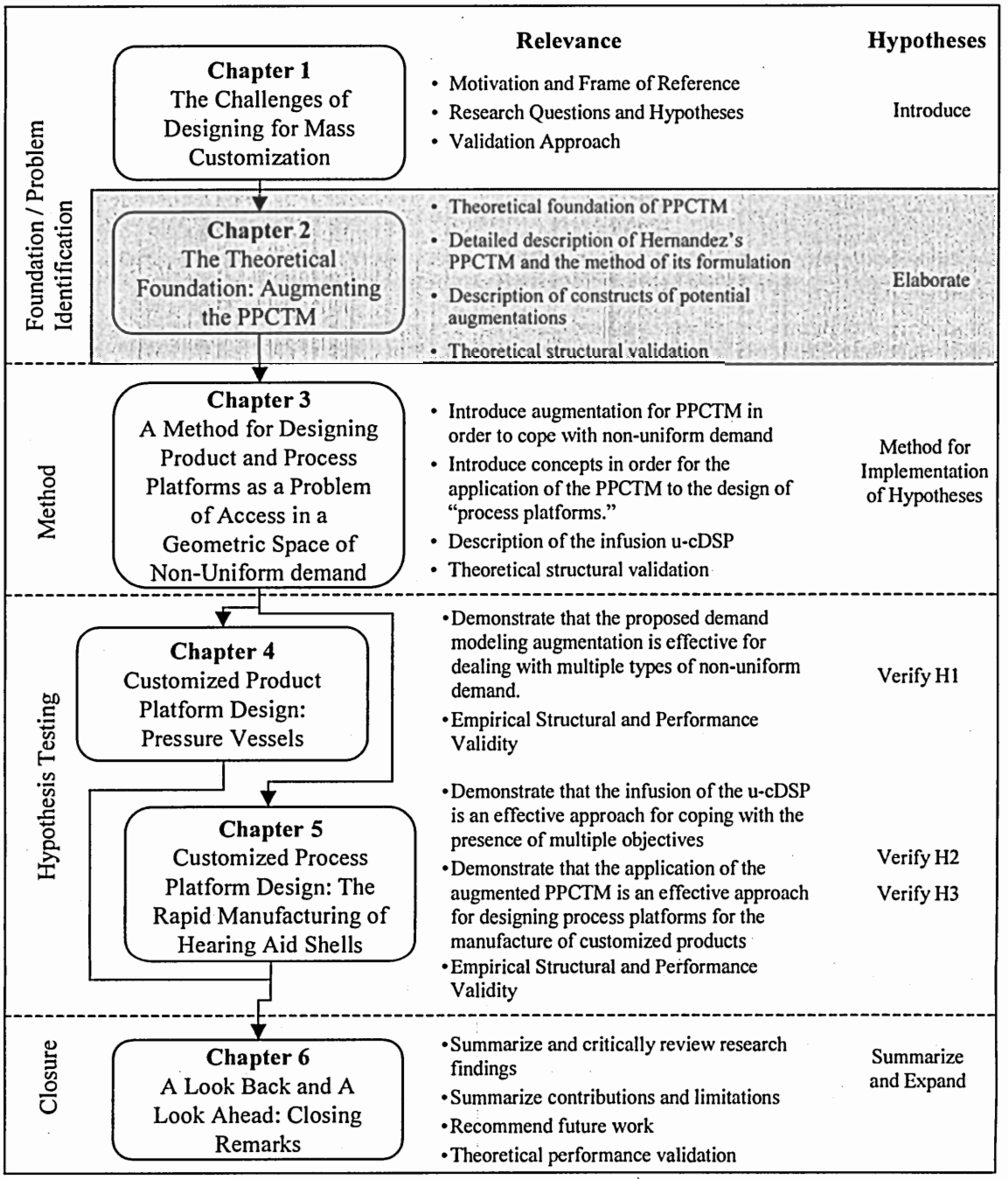


Figure 2.1 – The Role of Chapter 2 in this Thesis

2.1 PRODUCT PLATFORM DESIGN AS A PROBLEM OF ACCESS IN A GEOMETRIC SPACE

The focus in this thesis is to augment the Product Platform Constructal Theory Method (PPCTM). Before augmentations are presented, the details of history, concepts, and formulation of the PPCTM must first be presented. As such, the role of this section is to provide these specific details of the PPCTM. In Section 2.1.1, an account of the two theoretic underpinnings of his work, hierarchic systems theory and constructal theory, are discussed. A summary of the PPCTM's six-step method is then provided in Section 2.1.2. Each step of the method is illustrated with the running tutorial example of developing a product platform for a line of customizable cantilever beams. The entirety of this section is based on the work presented by Hernandez in his doctoral dissertation (Hernandez 2001) and is cited where appropriate.

2.1.1 Theoretic Underpinnings: Hierarchic Systems Theory and Constructal Theory

The fundamental problem addressed in (Hernandez 2001) is how to determine and organize *modes for managing product variety*– any generic approach in product design or its manufacturing process for achieving a product customization – systematically in order to create a product platform for customized products. Common modes for managing product variety include, but are not limited to, modular design, platform design and standardization, robust design, dimensional customization, and adjustable customization. These modes are the linking mechanism between the individual products that compose a product family. Previous work in the design of product platforms (see Section 1.2.3) has only focused on the development of a single mode for managing product customization.

Hierarchic Systems Theory

Hernandez's approach for organizing common components for a very large number of product variants is anchored in the thesis of Herbert Simon, who observed that complex structures adapt and evolve more efficiently when they are organized hierarchically (Simon 1996). Specifically, Simon and Ando made two fundamental observations regarding an apparent tendency in the natural self-organization of complex systems (Simon and Ando 1961):

1. Complexity, both in natural and artificial systems, frequently takes the form of a hierarchy, whereby *a hierarchic system is defined as being composed of interrelated subsystems that have in turn their own systems, and so on, until some elementary level of components is reached.*
2. In general, interactions inside subsystems (in a hierarchically-organized system) are stronger and/or more frequent than those interactions that occur between subsystems.

One can picture an artificial hierarchic system through the visualization of a typical company workforce. In this scenario there exists a hierarchy of authority: employees are typically supervised by manager who, in turn, is subordinate to another person. This pattern repeats itself until the "parent" (in this case, the CEO) of the hierarchy is reached. A graphic example of a hierarchic system, and the counter example of a non-hierarchic system, is provided in Figure 2.2.

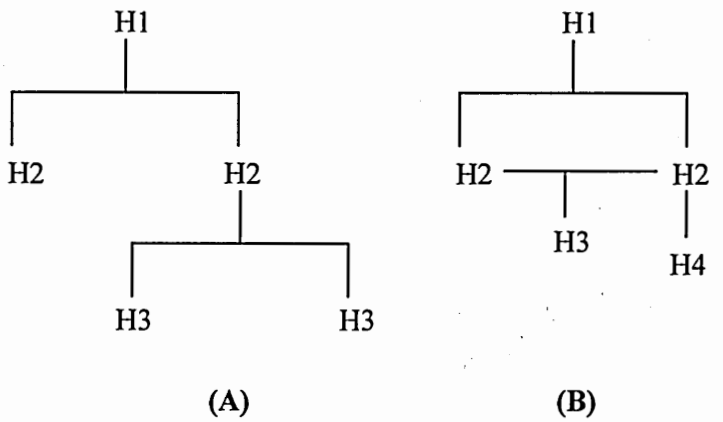


Figure 2.2 – Hierarchic (A) vs. Non-Hierarchic (B) Organization of Systems

Simon and Ando also presented the concept of near-decomposability (Simon 1996). This captures the idea that when organizing complex systems hierarchically, favorable conditions arise when each subsystem operates nearly independently of the processes happening within the other subsystems.

The concepts of hierarchic organization and near-decomposability are two fundamental posits of the PPCTM (Hernandez 2001):

> *Posit 1:* Potential for rapid adaptation and/or response is higher in complex systems when they are organized hierarchically.
> *Posit 2:* In hierarchically organized systems, the high-frequency (short run) responses tend to be associated with the lowest levels of hierarchy and the low-frequency (long run) ones with the interactions of these subsystems, i.e., the higher levels of the hierarchic organization.

With the idea of organizing the modes of managing product customization as a hierarchy, Hernandez searched for a construct in which the modes could be efficaciously organized.

Constructal Theory

Hernandez's approach to organize modes of managing product customization in a hierarchic manner is based on the representation of platform design as a problem of optimization of access in a geometric space. An optimal access problem is characterized by the need to determine the optimal "bouquet of paths" that link all points of an area S with a common destination O (Figure 2.3).

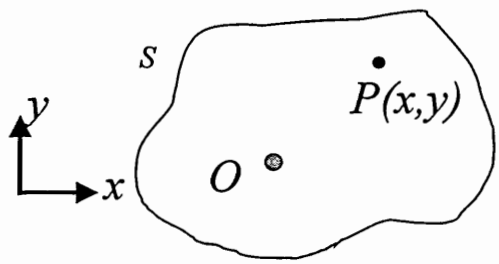


Figure 2.3 – A Finite-Size Area with a Common Destination (Bejan 1997)

The arrangement of these paths is usually optimized towards some objective (i.e, most efficient, least resistance, minimal travel time).

Constructal theory was initiated by Adrian Bejan as a result of studying problems of optimal access (Bejan 1996). Bejan's constructal theory rests on the premise that a single principle – the constrained optimization of global performance – is the generating mechanism of organization, complexity, and hierarchic structure in nature, engineering, and even management.

The crux of contructal theory is the concept that hierarchic organizations observed in nature are the result of a sequential optimization process with the objective of the maximization of access (or minimization of resistance, or losses). Following the basic tenants of constructal theory, this optimization process should proceed in a specific time direction: from the optimization of the basic elements to the higher-order assemblies of the structure. For example, the formation of tree networks observed in nature (such as blood vessels, tree branches, river tributaries, etc.) is explained as a hierarchic process of optimization: the shape that optimizes "access" at the most elementary volume occurs first, followed by an assembly of these innermost spaces into a second, larger, shape, which in turn are assembled into a third space, and so on. This sequential process continues until each space is connected and the entire space is covered, as shown in Figure 2.4.

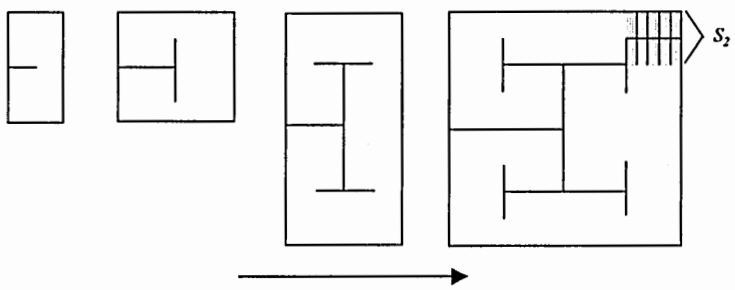


Figure 2.4 – Organization in Nature as a Problem of Access Optimization (Bejan 2000)

A common example of the application of constructal theory is the formation of a street network as a hierarchic process of optimization. Referring to Figure 2.3, the example entails a land of area S with O as a central market. The problem is how to link every point on the land ($P(x,y)$) to this central market. While the obvious solution is to create a straight-lined street to every point from O, this is not the optimal solution if there exists more than one mode of transportation. In this example there exists a faster mode of transportation (i.e., velocity of driving an automobile, V_1, compared to the velocity of walking, V_o), and there exists a limitation on the areas on which some modes cannot travel (i.e., there must be areas where the road does not cross in order to create a habitable space). As such, the solution then is to bring a finite number of streets near patches of land where inhabitants can access the streets (in order to travel at a velocity, V_1) via walking (V_o).

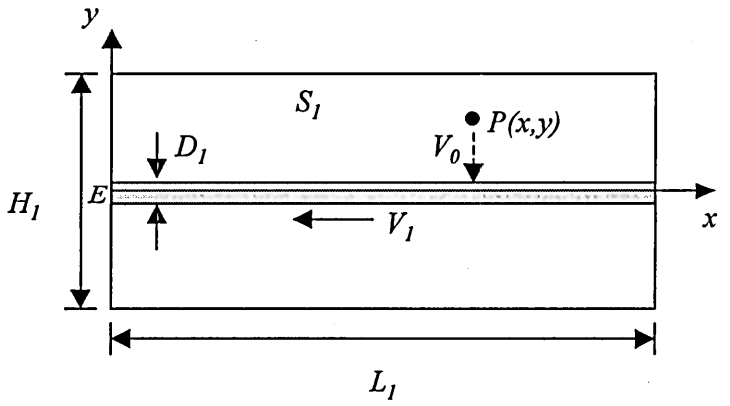


Figure 2.5 – The First Area Element (Bejan 1997)

Fundamentally, the access problem of this example is allocating a finite length of street to each finite patch of area, and also connecting these street lengths in an optimum manner such that travel time is minimized.

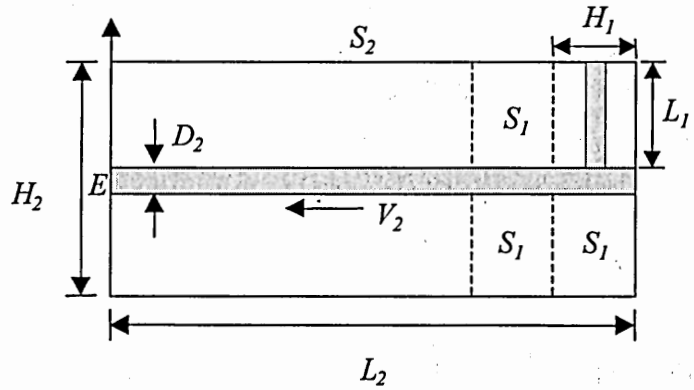


Figure 2.6 – The Second Area Element (Bejan 1997)

The direction of the solution by constructal theory is to assemble geometrically optimized areas (from smallest to largest) until the entire area, *S*, is covered as shown schematically in Figure 2.4.

It has been shown, however, that such this sequential optimization process yields sub-optimal results (Hernandez 2001). Bejan's constructal theory, however, provides the idea of formulating problems of access in geometric spaces as multi-stage optimization problems for which more effective solution algorithms can be utilized.

The outcome of Hernandez's research of constructal theory was three additional posits to the PPCTM (Hernandez 2001):

> *Posit 3:* Systems complexity results from a natural process of systems to provide paths of easier access.
>
> *Posit 4:* Each path of access within the optimized system structure is unique and does not cross with other paths: the resulting structure is hierarchic.
>
> *Posit 5:* The design of a hierarchic structure to provide easier access should proceed in a specific time direction: from the optimization of the basic elements at the smallest scale towards the optimal arrangement of these elements into higher-order assemblies, the process being one of repeated maximization of access (or minimization of losses) subject to constraints.

With its theoretic foundation established, the Product Platform Constructal Theory Method itself is presented next.

2.1.2 The Product Platform Constructal Theory Method

Hernandez's application of constructal theory in the PPCTM begins with the definition of a *space of customization*. A space of customization is the set of all feasible combinations of values of product specifications that a manufacturing enterprise is willing to satisfy (Hernandez, Allen et al. 2002); it is the equivalent of the geometric space one wishes to optimize in the street network example discussed above.

In the creation of a product family for customized products, a designer wishes to link all different feasible products within the space of customization from a common starting point. The manner in which each possible product is linked when using the PPCTM is through the modes for managing product variety that were defined in Section 2.1.1. The use of "modes for managing product variety" is similar to the use of streets in the street network example presented above; they both provide a means for linking the fundamental sub-spaces. This similarity is shown graphically in Figure 2.7. The point *O* represents the baseline product, the separate points, *P*, represent different product variants, and the modes for managing variety are shown as *ΔP*.

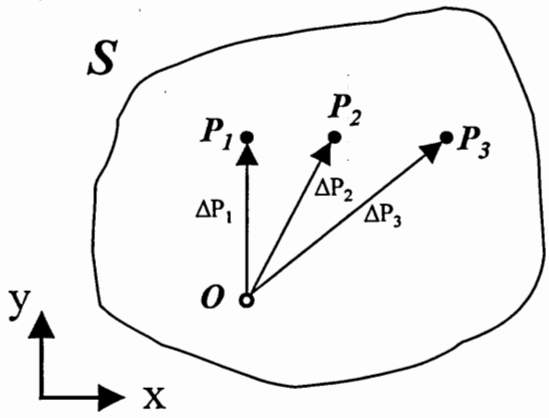


Figure 2.7 – Product Platform Design as a Problem of Optimal Access

Table 2.1 is provided as another example of the analogy that exists between the traditional constructal theory example, street network design, and the development of product platforms for customizable products.

Table 2.1 – The Analogy of Creating a Street Network and Designing Product Platforms via the Application of Constructal Theory

	Street Network	**Product Platform**
Geometric Space	City	Space of Customization
Crux of Problem	Connecting all points in city together to ensure optimal access	Connecting all product variants in customization space to a "baseline" to ensure minimum cost
Means of linking spaces	Streets	Modes of managing product customization
Solution Goal	Optimize geometric shape of each space element	Select the optimum range of each mode of managing product customization across the entire space
Objective	Minimizing travel time	Maximizing profit

The problem of designing a platform for customizable products becomes an effort to define a baseline set of components (the product platform) from which one can access all the points of a space of customization through the systematic use of a series modes for managing product customization, and optimizing some given objective (e.g., cost, profit, customer service, etc). Following hierarchic systems theory, Hernandez proposed to solve this access problem through the hierarchic application of modes for managing product variety. For the determination of the range of application for each mode, Hernandez looked to the application of Bejan's constructal theory. Figure 2.8 is provided as an illustrative example of this concept. In the context of this figure, the role of the application of constructal theory to this problem is the organization of the modes for managing product variety that offer the most efficient manner of generating product variants. Similar to the street network problem presented above, it is not efficient to develop a mode for managing variety for each product variant (as shown in Figure 2.7).

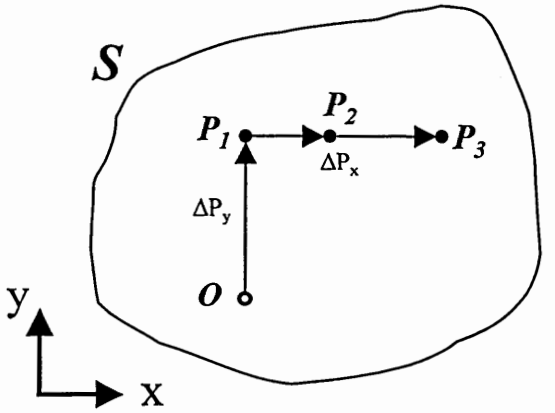


Figure 2.8 – The Use of Constructal Theory in Product Platform Design

Based on constructal theory, and the abstraction of platform design as a problem of access in a space of customization, Hernandez formulated the design of platforms for customizable products as a multistage optimization problem. This problem is then solved by dynamic programming (discussed in some detail in "Step 5: Formulate the Multistage Problem as a Dynamic Program" below).

The result of Hernandez's research is the Product Platform Constructal Theory Method, a six-step method for which the design of product platforms for customizable products is made possible (Figure 2.9). Each of these six steps is described in detail in the following sub-sections. The running tutorial example of customizable cantilever beams is presented in order to assist with the illustration of the method.

Step 1: Define the Market Space

In the first step of Hernandez's method, the space of customization must first be defined. This consists of two steps: identifying the product parameters that will be varied according to customer demands, and identifying the range of variety that will be offered according to the market demand. The number of product parameters that will experience variety determines the dimension of the space. The bounds of the geometric space to be

analyzed are defined by the ranges of the parameters for which variety will be offered. This range is usually bounded by economic or technological limitations.

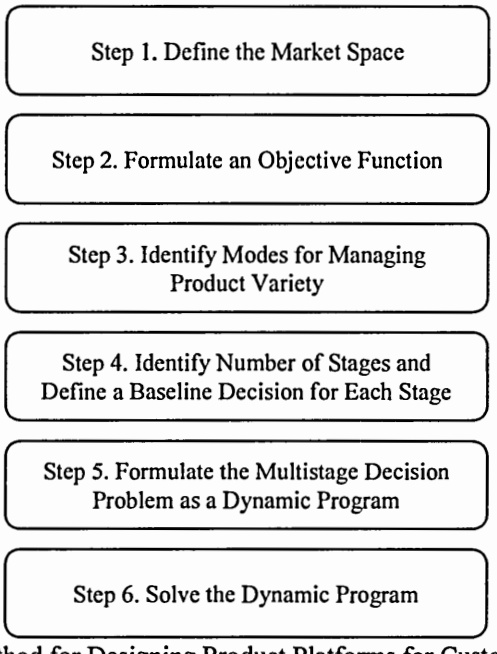
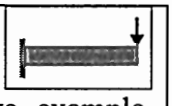


Figure 2.9 – Hernandez's Method for Designing Product Platforms for Customizable Products (Hernandez 2001)

Box 2.1 – Defining the Market Space of the Cantilever Beam Tutorial Example



Consider a line of customizable cantilever beams as an illustrative example. CutomTech, a company specializing in the offering of customizable products, believes that offering a line of customizable cantilever beams would provide them with a significant competitive advantage over their competitors. The embodiment design of their cantilever beam is presented in Figure 2.10.

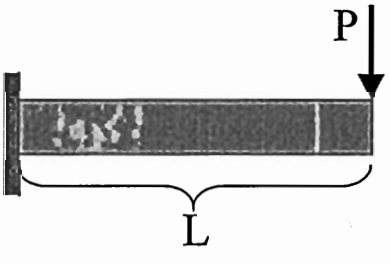


Figure 2.10 – Customizable Cantilever Beam

The design parameters involved with the development of a cantilever beam are:

- E: the modulus of elasticity (MPa); a material property
- σ_u: ultimate strength (MPa); a material property
- I: moment of inertia (mm^4); a property of the cross-section
- c: largest distance from the neutral surface (mm); a property of the cross-section

The customers of CustomTech are mainly concerned with two specifications of cantilever beams: the length (m) and the maximum applied load it can withstand before yielding (N).

Marketing research of CustomTech shows that there exists a large demand for a wide variety of these parameters. The range of length demanded is from 1 to 3 meters. The range of maximum end load demanded is from 1 to 5 N. The resulting space of customization of this example is a two-dimensional bounded space as shown in Figure 2.11, where each point in this geometric space represents a unique combination of the cantilever beam two specifications: length and maximum end load.

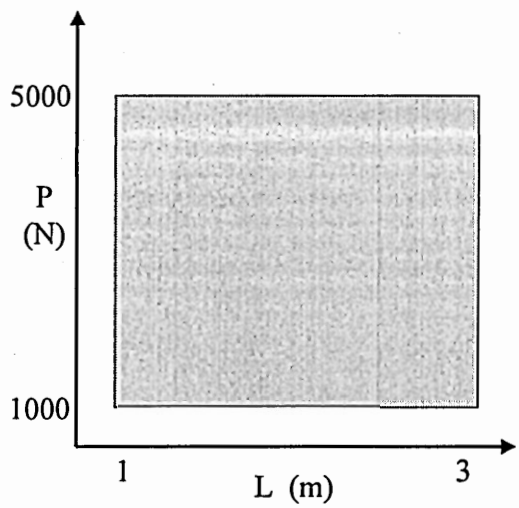
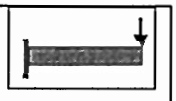


Figure 2.11 – Space of Customization for Cantilever Beam Example

Step 2: Define the Objective Function

The second step of the PPCTM features the identification of the parameter to be improved by the method. Some examples of appropriate objective functions for the development of product families include, the minimization of the average cost, or the maximization of the average profit; any other suitable design parameter (i.e., quality, or performance) is acceptable.

Box 2.2 – Defining the Objective Function for the Cantilever Beam Tutorial Example



Returning to the tutorial example, assume that the manufacturer, CustomTech, wishes to minimize the average cost of the entire product family. The manufacturer could also choose to optimize other metrics of the product family, i.e., minimize mass, minimize deflection, minimize radius of curvature, or maximize profit. For this tutorial example the cost of an individual cantilever beam is represented as a simple function of length, neutral surface distance, moment of inertia, and ultimate strength, $C(L, c, I, \sigma_u)$.

To evaluate the cost of the entire market space, the cost function must be first defined in terms of the two variable product parameters (maximum end load, P, and length, L) in order to integrate it over the market space:

$$c = \int_{L_{min}}^{L_{max}} \int_{P_{min}}^{P_{max}} C(P,L)\,dP\,dL \qquad [2.1]$$

If the preceding example were taken for a market space of discrete values, the objective function would simply be calculated as the sum of the cost of all the product variants in the entire space:

$$c = \sum_{j=P_{min}}^{P_{max}} \sum_{i=L_{min}}^{L_{max}} C_{i,j}(P,L) \qquad [2.2]$$

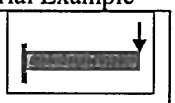
For the PPCTM's working assumption of uniform demand, the average cost is simply evaluated as the quotient of Equation 2.1 and the demand, D.

$$c = \frac{1}{D} \int_{L_{min}}^{L_{max}} \int_{P_{min}}^{P_{max}} C(P,L)\,dP\,dL \qquad [2.3]$$

Step 3: Identify the Modes for Managing Product Variety

Once a market space is identified and an objective function is defined, the next step of the PPCTM is to consider how to vary a product concept in order to satisfy all the customized specifications of the market space. Discussed earlier in Section 2.1.1, modes for managing product variety are the linking mechanism between the products that compose the product family. This step involves collaborative brainstorming and decision-making between design and manufacturing.

Box 2.3 – Identifying the Modes for Managing Product Variety for the Cantilever Beam Tutorial Example



The designers of the customizable line of cantilever beams propose three modes of managing product customization:

1) *Customization of beam length*: To minimize costs in the manufacturing process, it is decided to order raw lengths of the beam material from which the customer's desired length is cut-to-order. This allows the manufacturer to offer any length of beam within the continuous range previously defined. While this minimizes the cost associated with ordering, it will add a cost associated with waste. This mode offers variety along the axis of length.
2) *Standardize cross-section*: The cross-section of the beam is standardized in order to

minimize costs associated with ordering. This mode offers variety along the axis of maximum applied load.

3) *Standardize material*: The standardization of material offers variety along the axis of maximum applied load. This mode also offers an opportunity to minimize cost through discounts in ordering.

Step 4: Identify Number of Stages and Define a Baseline Decision for Each Stage

With the modes for managing product variety identified, the method in which they are utilized must be established. Typically the application of a mode depends on the magnitude and frequency of changes in product variety desired. From Hernandez's second posit (Section 2.1.1) it is established that the modes capable of achieving the smallest variations in design parameters should be used first. Some approaches, such as dimensional customization, can be used to satisfy very small variations in the product, but are typically expensive for large changes. Other approaches, such as modularity, are not economically feasible for small changes. As such, economical and technological considerations place an important role in establishing the hierarchic use of the modes for managing product variety.

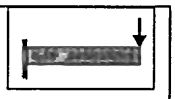
Once the order of the use of the modes for managing product variety is established, the baseline decision to be made at each stage must be formulated. This decision is simply the selection of the design parameters that minimize (or maximize) the given objective function for that specific stage. A generic baseline decision formulation is provided in Figure 2.12.

	For Stage *i*
Given	The N-dimensional market space $M^N = \{(r_1, r_2, ..., r_N)\}$
	The decision variables of previous stages $\Delta r(1), ..., \Delta r\ (i\text{-}1)$
	The modes for managing product variety to be utilized at Stage i
	The set of constraint functions $g[\Delta r(1), ..., \Delta r\ (i\text{-}1)] \geq 0$
	An objective function $f_i\ [\Delta r(1), ..., \Delta r\ (i)]$
Find	The value of the decision variables $x(i) = [\Delta r_1(i), \Delta r_2(i), ..., \Delta r_N(i)]$
Satisfy	*Constraints* $\Delta r_j(i) \geq \Delta r_j(i-1)$
	Bounds $\Delta r_{j\min} \leq \Delta r_j(i) \leq \Delta r_{j\max}$
Optimize	The value of the objective function for that stage
	$f_i = f_i[\Delta r(1), ..., \Delta r(i)]$

Figure 2.12 – Baseline Decision for an Arbitrary Stage *i* (Hernandez 2001)

There are constraints on the decision variables such that the increment in each design variable is not greater than the corresponding increment for the previous stage. Bounds exist in the decision to prevent the decision variable increment from being smaller than itself, or larger than the amount of variability desired.

Box 2.4 – Identifying the Number of Stages and Defining a Baseline Decision for Each Stage of the Cantilever Beam Tutorial Example



In this step in the cantilever beam example, it is required to order the previously proposed modes of managing customization in a hierarchy. The order of the modes, and their resulting baseline decisions are presented below:

<u>First Space Element:</u> *Customization of Beam Length* (Mode L) – This mode offers the most cost effective manner to offer the largest variability.

For Stage 1	
Find	The value of the decision variable ΔL
Satisfy	*Bounds* $\Delta L \leq L_{max}$
Optimize	The value of the objective function for this stage: $$c_1 = \frac{1}{D}\int_{L_{min}}^{L_{max}}\int_{P_{min}}^{P_{max}} C(P, L_1)\,dP\,dL$$

Figure 2.13 – Baseline Decision for Stage 1 of Cantilever Beam Example

Fundamentally, the crux of this decision is the identification of the range for which the raw beam length will be commonalized (ΔL).

Second Space Element: *Standardization of Cross-Section* (Mode P_1)– This mode offers the most amount of variability in the maximum load direction.

For Stage 2	
Given	The value of ΔL
Find	The value of the decision variable ΔP_1
Satisfy	*Bounds* $\Delta P_1 \leq P_{max}$
Optimize	The value of the objective function for this stage: $$c_2 = \frac{1}{D}\int_{L_{min}}^{L_{max}}\int_{P_{min}}^{P_{max}} C(P_1, L)\,dP_1\,dL$$

Figure 2.14 –Baseline Decision for Stage 2 of Cantilever Beam Example

Fundamentally, the crux of this decision is the identification of the range for which the cross-section of the beam will be commonalized (ΔP_1).

Third Space Element: *Standardization of Material* (Mode P_2) – The final mode offers the smallest amount of variability and at the most expense.

For Stage 3	
Given	The value of ΔL, ΔP_1
Find	The value of the decision variable ΔP_2
Satisfy	*Bounds* $\Delta P_1 \leq \Delta P_2 \leq P_{max}$
Optimize	The value of the objective function for this stage: $$c_3 = \frac{1}{D}\int_{L_{min}}^{L_{max}}\int_{P_{min}}^{P_{max}} C(P_2, L)dP_2 dL$$

Figure 2.15 – Baseline Decision for Stage 3 of Cantilever Beam Example

Fundamentally, the crux of this decision is the identification of the range of the commonality of the material choice (ΔP_2).

Step 5: Formulate the Multistage Decision Problem as a Dynamic Program

As discussed above, the baseline decisions of each stage established in the previous step are formulated as a multistage optimization problem in the implementation of the PPCTM. The fifth step of the PPCTM suggests the use of dynamic programming to solve this complex problem. Dynamic programming is an approach for optimizing multistage decision processes and is based on the concept that the optimal parameters for the global problem are based on the selection of the parameters through the knowledge of the optimal parameters at each stage of the decision problem (Jacobs 1967). It is based on Bellman's principle of optimality:

> *Principle of Optimality*: An optimal policy in a multi-stage optimization process has the property that regardless of the decisions taken to enter a particular stage, the remaining decisions must constitute an optimal policy for leaving that state.

Dynamic programming begins with the last stage of an *n*-stage process and determines for each state the best policy for leaving the stage and completing the process,

assuming that all preceding stages have been completed. The decision maker then moves "backwards" through the process stage by stage.

Formulating the multi-stage problem as a dynamic program involves:

1) defining the optimal objective function for each stage
2) formulating the general recurrence equation.
3) Developing a table or closed-form equation for the optimal objective function for each stage
4) Formulate the decision to be made at each stage of the solution process

The formulation of the multistage problem as a dynamic program transforms Figure 2.12 into Figure 2.16:

	For Each Stage *i*
Given	The N-dimensional market space $M^N = \{(r_1, r_2, ..., r_N)\}$
	The decision variables of previous stages $\Delta r(1), ..., \Delta r(i-1)$
	The modes for managing product variety to be utilized at Stage i
	The set of constraint functions $g[\Delta r(1), ..., \Delta r(i-1)] \geq 0$
	An objective function $f_i[\Delta r(1), ..., \Delta r(i)]$
Find	The value of the decision variables $x(i) = [\Delta r_1(i), \Delta r_2(i), ..., \Delta r_N(i)]$
Satisfy	*Constraints* $\Delta r_j(i) \geq \Delta r_j(i-1)$
	Bounds $\Delta r_{j\min} \leq \Delta r_j(i) \leq \Delta r_{j\max}$
Optimize	The value of the objective function for that stage
	$f_i^* = f_i[\Delta r(1), ..., \Delta r(i)] + f_{i+1,k}^*[\Delta r(1), ..., \Delta r(i)]$

Figure 2.16 – Formulation of the Multi-Stage Problem as a Dynamic Program (Hernandez 2001)

where the variable f_i^* is the new objective function for any stage *i*.

Box 2.5 – Formulating the Multistage Decision Problem of the Cantilever Beam Tutorial Example as a Dynamic Program

There are three decision stages in the cantilever beam example which are formulated into an overall multi-stage formulation:

Find	The value of the decision variable ΔL, ΔP_1, ΔP_2	
Satisfy	*Bounds*	$\Delta L \le L_{max}$
		$\Delta P_1 \le P_{max}$
		$\Delta P_1 \le \Delta P_2 \le P_{max}$
Optimize	The objective function (same as the third stage):	

$$c_3 = \frac{1}{D}\int_{L_{min}}^{L_{max}}\int_{P_{min}}^{P_{max}} C(P_2, L)\,dP_2\,dL$$

Figure 2.17 – The Multistage Problem for the Cantilever Beam Example

Step 6: Solve the Dynamic Program

In the final step of the PPCTM, the dynamic program established in the previous step is solved. The solution can involve the development of response surface in order to approximate optimal objective functions for each stage by moving "backwards" through the problem. The solution of the problem is then carried out in a forward manner through the use of these response surfaces, as shown in Figure 2.18.

As can be seen in the Figure 2.18, the process for approximating optimal objective functions using the response surface method involves a multi-step process (from (Hernandez 2001):

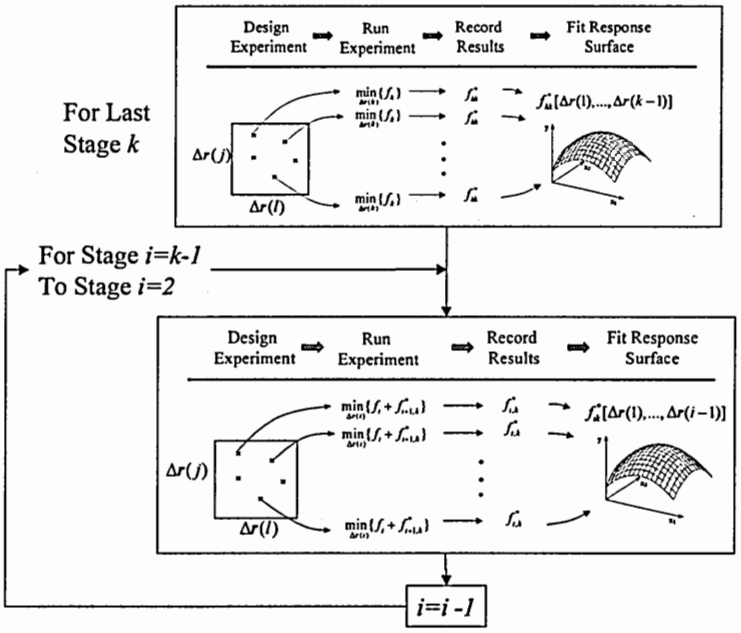


Figure 2.18 – Developing Response Surface Approximations for Optimal Objective Functions in Dynamic Programming (Hernandez 2001)

1. For the last stage k of the process:

   a. An experiment is designed with the value of the decision variables from previous stages, $\Delta r(1)$, $\Delta r(2)$, …, $\Delta r(k\text{-}1)$ as independent factors and the value of the optimal objective function, f_k as the response variable.

   b. For each run (i.e., each combination of the decision variables $\Delta r(1)$, $\Delta r(2)$, … $\Delta r(k\text{-}1)$) from the experimental design, the value of f_k is optimized by choosing the appropriate value of $\Delta r(k)$ and the value of the optimized function obtained, referred to as f_{kk}^{*}, is recorded.

   c. After all runs are completed, a response surface is fit to the optimized values of f_{kk}^{*} (recorded from the previous step) as a function of the decision variables of previous stages $\Delta r(1)$, $\Delta r(2)$, … $\Delta r(k\text{-}1)$. This

response surface is an approximation of the optimal objective function for Stage k.

2. For Stage $i = k$-1 to Stage $i = 2$:

   a. An experiment is designed with the value of the decision variables from previous stages, $\Delta r(1)$, $\Delta r(2)$, … $\Delta r(i\text{-}1)$ as independent factors and the value of $f_i + f^*_{i+1,k}$ as the response variable, where the optimal objective function, $f_i + f^*_{i+1,k}$, has been obtained in the preceding step.

   b. For each run (i.e., each combination of the decision variables $\Delta r(1)$, $\Delta r(2)$, …, $\Delta r(i\text{-}1)$) from the experimental design, the value of $f_i + f^*_{i+1,k}$ is optimized by choosing the appropriate value of $\Delta r(i)$ and the value of $f_i + f^*_{i+1,k}$ obtained is recorded. This value is referred to as $f^*_{i,k}$.

   c. After all runs are completed, a response surface is fit to the values of $f^*_{i,k}$ recorded from the previous step as a function of the decision variables of previous stages $\Delta r(1)$, $\Delta r(2)$, …, $\Delta r(i\text{-}1)$. This response surface is an approximation of the optimal objective function for Stage i-1.

   d. $i = i - 1$ and steps (a) through (c) are repeated.

The solution of this dynamic program results in the selection of optimal ranges of application for each mode for managing product variety. These values can be then be used to map the design parameters to a fixed set of components that can be varied using the modes of managing variety in order to meet individual customer specifications.

Box 2.6 – Solving the Dynamic Program of the Cantilever Beam Tutorial Example

Due to the simplicity of the cantilever beam example, an exhaustive search of the combinations of the three decision variables is sufficient for the solution of this problem. The resulting three-loop iteration of the variables is shown in Figure 2.19.

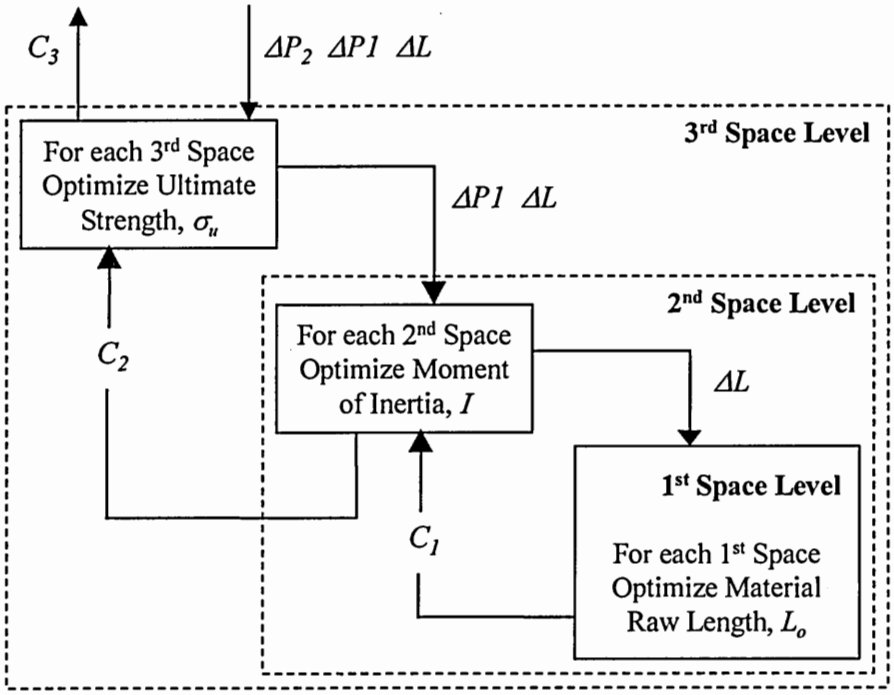


Figure 2.19 – The Solution Algorithm for the Cantilever Beam Example

The result of this solution algorithm is the determination of the ranges of each mode for managing variety that provides the lowest average cost of the entire product platform. Illustrative results are provided in Table 2.2.

Table 2.2 – Illustrative Results of Cantilever Beam Tutorial Example

Decision Variable	*Mode of Managing Variety*	*Range of Commonality*
ΔL	Customization of Length of Beam	0.33 m
ΔP_1	Standardization of Cross Section	667 N
ΔP_2	Standardization of Material	2000 N

These results inform the designers at CustomTech that by commonalizing the raw beam length across a range of 0.33 m, standardizing the cross section across a range of 667 N, and standardizing the material type across a range of 2000 N, the cost to manufacture the entire product platform will be at a minimum. The resulting hierarchical arrangement of the three modes of managing product variety is shown in Figures 2.20 and 2.21. The arrangement of each subspace in the original space of customization (Figure 2.11) is shown in Figure 2.20.

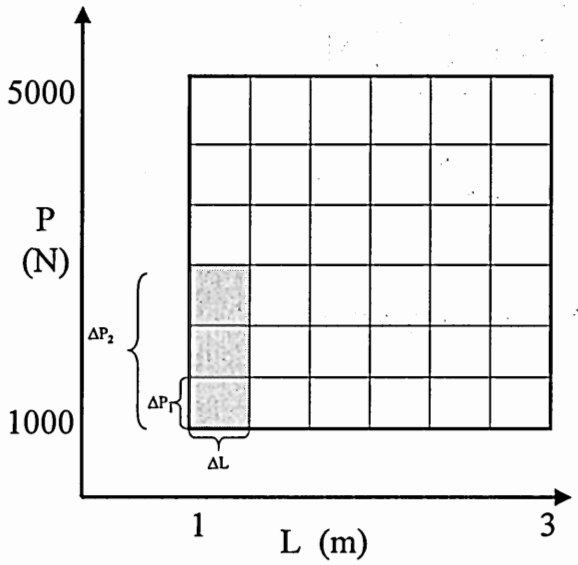


Figure 2.20 – Sample Hierarchic Solution of the Cantilever Beam Example

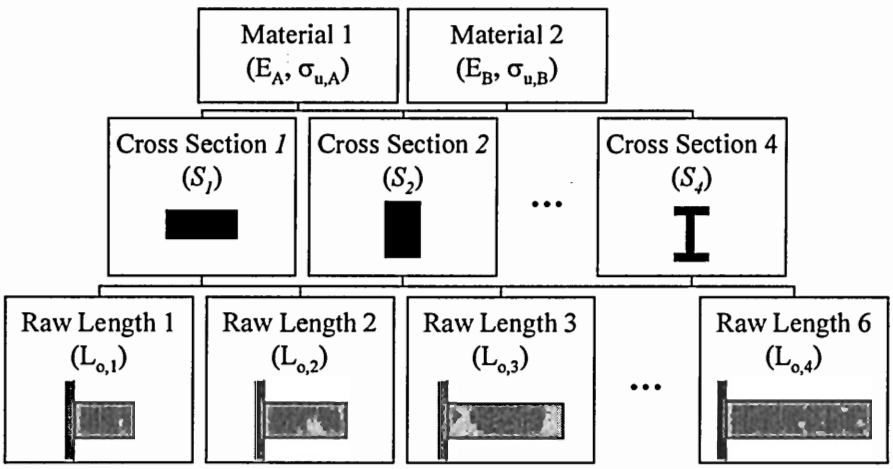


Figure 2.21 – Hierarchical Arrangement of the Modes of Managing Product Variety for the Cantilever Beam Tutorial Example

Using the optimal arrangement and optimal ranges for each mode of managing variety, the specific design parameters for each product variant can be mapped, as shown in Table 2.3.

Table 2.3 – Mapping Modes of Managing Variety to Product Specifications

L (m)	*P (N)*	*Lo (m)*	*Cross Section*	*Material Type*
1.33	2000	Lo,1	CS,1	Material 1
	3000		CS,2	
	4000		CS,3	Material 2
	5000		CS,4	
1.67	2000	Lo,2	CS,1	Material 1
	3000		CS,2	
	4000		CS,3	Material 2
	5000		CS,4	
2	2000	Lo,3	CS,1	Material 1
	3000		CS,2	
	4000		CS,3	Material 2
	5000		CS,4	
2.33	2000	Lo,4	CS,1	Material 1
	3000		CS,2	
	4000		CS,3	Material 2
	5000		CS,4	
2.67	2000	Lo,5	CS,1	Material 1
	3000		CS,2	
	4000		CS,3	Material 2
	5000		CS,4	
3	2000	Lo,6	CS,1	Material 1
	3000		CS,2	
	4000		CS,3	Material 2
	5000		CS,4	

The results posted in the above figure provide CustomTech with a roadmap of design specification of the entire product family. For example, assume that a customer requests a custom cantilever beam of length 215 in. that can support a load of 3500 lbs. The results posted in Table 2.3 inform CustomTech that this need can be met by cutting a beam of cross-section of CS,3 and material #2 from a raw length of $L_{o,4}$.

2.1.3 Critical Evaluation of the Product Platform Constructal Theory Method

The Product Platform Constructal Theory Method is based on the foundation of hierarchic systems theory and constructal theory. The concepts of hierarchic systems theory state that a hierarchic method of organization is the most efficient for complex systems. The infusion of constructal theory provides the concept of formulating problems of access in geometric spaces as multistage optimization problems. The resulting six-steps of the PPCTM successfully synthesize multiple modes of managing product variety for the development of a product platform for customized products. The resulting hierarchic organization of the modes of managing product variety provides a designer with the optimal range of commonality for each mode without sacrificing performance. This list of successes fulfill the limitations of those product platform methods previously reviewed (Section 1.2.3), and also satisfied Hernandez's research goals (Hernandez 2001).

Some limitations of the PPCTM include:

(1) *Unable to deal with multiple objectives*: As systems become more complex, multiple, coupled, and conflicting goals become more prevalent. The examples presented thus far by Hernandez focus on the optimization of one objective. While the method, at its most abstract form, is easily adaptable to the solution of multiple objectives, it has yet to be shown that this is possible.

(2) *Focus on optimization*: Related to the previous limitation is Hernandez's focus on optimization. This serves as a minor limitation; a designer with multiple objectives cannot search for an "optimal" solution. Therefore, a tool is needed to

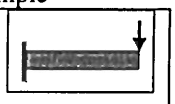
capture the notion of searching for a solution that best satisfies all of the goals and their inherent tradeoffs.

(3) *Unable to deal with non-uniform demand*: The demand distributions of the market spaces of mass-customization are as varied as the individual products produced within them. The examples presented thus far by Hernandez (pressure vessel (Hernandez, Allen et al. 2002), beverage merchandiser (Hernandez 2001), electric motors (Hernandez, Allen et al. 2003)) do not cope with non-uniform demand. Although Hernandez claims that the definition of his method is open enough to accept the modeling of non-uniform demand, no example has been presented to verify this.

(4) *No consideration for the manufacturing process*: Admittedly, this was out of the scope of Hernandez's earlier research. This can no longer be ignored, however, as more and more importance is being placed on the concurrent engineering of both the design of the product and process of customized products.

The severity of these limitations is illustrated in the cantilever beam tutorial example problem in Box 2.7.

Box 2.7 – The Impact of the Limitations of the PPCTM on the Cantilever Beam Tutorial Example



The use of the PPCTM provides CustomTech with a methodical manner to synthesize multiple modes of managing product variety. The result of the implementation of this design method also provides a map of the design specifications across the product platform. The designers of CustomTech are concerned with the results

of the application of the PPCTM due to its inability to model the entire problem of offering a customized cantilever beam.

For example, the demand for cantilever beams is highly non-uniform. Market research concludes that there exists a normal distribution of demand across the market, with the majority of the customers requesting 200 in. cantilever beams that can support a load of 3000 lbs. Currently, the application of the PPCTM does not take into account this non-uniform demand.

Secondly, the designers at CustomTech wish to use the PPCTM to select ranges for the modes for managing product variety with two different objectives: to minimize production cost, and to minimize the maximum deflection of the beam. It is hoped that, as a result, CustomTech will be able to provide a line of customizable cantilever beams of high quality at a low cost. At its current stage, the PPCTM cannot be used to model the designers' multiple objectives.

While the design of the product platform has been completed through the application of the PPCTM, the manufacturers at CustomTech are concerned with the design of the production process. Specifically, they are concerned that their production process will be able to handle the wide variety of requirements for production capacity effectively. The current application of the PPCTM ignores the need to design the product and the production process concurrently.

As can be seen, CustomTech could benefit from an augmented PPCTM in which non-uniform demand, multiple objectives, and production process concepts can be modeled.

As stated earlier in Section 1.3, the focus in this thesis is the augmentation of Hernandez's PPCTM. The discussion of the tools for its augmentation is presented next.

2.2 THE UTILITY-BASED COMPROMISE DECISION SUPPORT PROBLEM

At the conclusion of the previous section, four limitations of the Product Platform Constructal Theory Method are identified. The first two listed, the inability to deal with multiple objectives and the focus on optimization, can be effectively addressed through the infusion of the utility-based compromise Decision Support Problem (u-cDSP).

"The u-cDSP provides structure and support for including human judgment in engineering decisions involving multiple attributes, while simultaneously providing an axiomatic basis for accurately reflecting the preferences of a designer with regard to feasible tradeoffs among these attributes under conditions of uncertainty (Fernández, Seepersad et al. 2001)." The formulation of the u-DSP is based on two constructs – the compromise DSP (c-DSP) and utility theory. Similar to the previous section, a running tutorial example of a line of customizable cantilever beams will be used to further illustrate the important details of the method.

2.2.1 Compromise Decision Support Problems

Compromise and selection Decision Support Problems (DSP) are general formulations for decision problems that are encountered in engineering design. The compromise DSP is a decision model for solving multi-objective, non-linear, optimization problems (Mistree, Smith et al. 1993). Mathematically the cDSP is a multi-objective decision model which is a hybrid formulation based on Mathematical

Programming and Goal Programming (Mistree, Hughes et al. 1993). It is used to determine the values of design variables, which satisfy a set of constraints and bounds and achieve as closely as possible a set of conflicting goals. The structure of the cDSP is as follows:

Given: A feasible alternative, assumptions, parameter values and goals.
Find: Values of design and deviation variables.
Satisfy: System constraints, system goals, and bounds on variables.
Minimize: Objective function (deviation variables that measure distance between goal targets and design points).

The system descriptors, namely, system and deviation variables, system constraints, system goals, bounds and the deviation function are described in detail in (Mistree, Hughes et al. 1993) and are not repeated here.

2.2.2 Utility Theory

Utility theory is a branch of decision theory in which a decision maker's preferences are assessed under conditions of risk and uncertainty. Its strength lies in the fact that it is based on rigorous mathematics and axioms. Overall, the method is able to *quantitatively* model designer preferences through the development of an expected utility function - a numerical function of an attribute's utility (Keeney and Raiffa 1976). If an appropriate numerical utility function can be developed, a designer can rank possible alternatives by calculating their expected utilities and then choosing the alternative with the highest expected utility. This, in turn, facilitates rational decision-making. This is significant when the designer is faced with a large number of objectives, for which simultaneous and ad hoc consideration is extremely difficult and the treatment of tradeoffs and uncertainty with respect to the attributes becomes crucial (Seepersad 2001).

2.2.3 Utility-Based Decision Support Problem

Seepersad augmented the cDSP with utility theory to create a multi-objective decision support model with an objective function derived from utility theory (Seepersad, Mistree et al. 2002). The fusion of the critical components of the two constructs is shown in Figure 2.22.

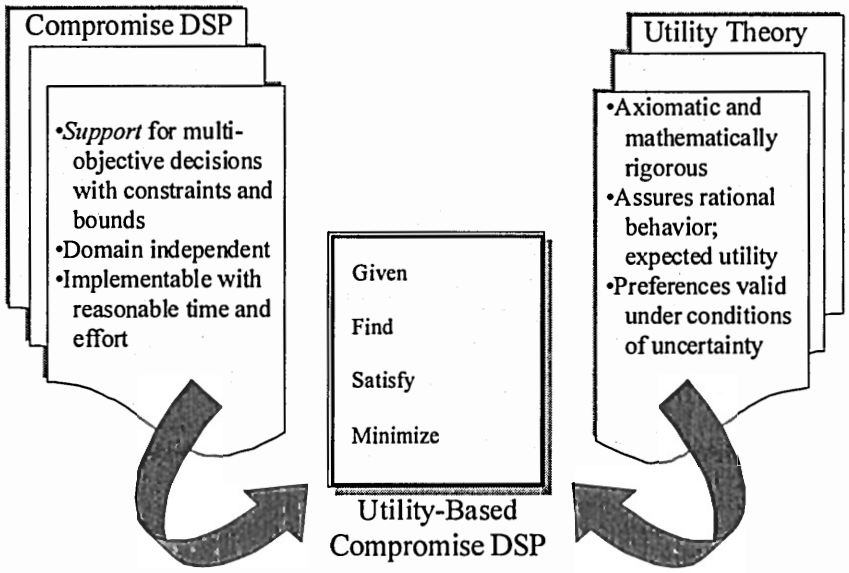


Figure 2.22 – Augmenting the Compromise DSP with Utility Theory to Form the Utility-Based Compromise DSP (Seepersad 2001)

The utility-based compromise DSP is a decision support construct that is based on utility theory and permits mathematically rigorous modeling of designer preferences such that decisions can be guided by expected utility in the context of risk or uncertainty associated with the outcome of a decision. The mathematical formulation of the u-DSP is provided in Figure 2.23.

This formulation is similar to that of the c-DSP, however the system goals and objective function are formulated using utility theory. The deviation function of the u-DSP is formulated to minimize deviation from the target expected utility (i.e., 1, the most preferable value), which is mathematically equivalent to maximizing expected utility.

Given

An alternative to be improved through modification.
Assumptions used to model the domain of interest.
The system parameters.
All other relevant information.

n	number of system variables
p+q	number of system constraints
p	equality constraints
q	inequality constraints
m	number of system goals
$g_i(\underline{X})$	system constraint functions
$A_i(\underline{X})$	system goals
$u_i(A_i(\underline{X}))$	utility function for each goal
$U(\underline{X})$	Overall, multi-attribute utility function $=f[u_1(A_1(\underline{X})), u_2(A_2(\underline{X})),\ldots, u_m(A_m(\underline{X}))]$

Find

The values of the independent *system variables* (that describe the physical attributes of an artifact).

$\underline{X}=X1, \ldots, Xj$ $\quad$ $j=1, \ldots, n$

The values of the *deviation variables* (that indicate the extent of which target utilities are achieved).

d_i^-, d_i^+ $\quad$ $i = 1, \ldots, m$

Satisfy

The *system constraints* that must be satisfied for the solution to feasible. There is no restriction placed on linearity or convexity.

$g_r(\underline{X}) = 0$ $\quad$ $r = 1, \ldots, p$

$g_r(\underline{X}) \geq 0$ $\quad$ $r = p+1, \ldots, p+q$

The *system goals* that must achieve a target utility to the extent possible. There is no restriction placed on linearity or convexity.

$E[u_i(A_i(\underline{X}))] + d_i^- - d_i^+ = 1;$ $\quad$ $i = 1, \ldots, m$

The lower and upper bounds on the system variables.

$X_j^{min} \leq X_j \leq X_j^{max};$ $\quad$ $k = 1, \ldots, n$

$d_i^-, d_i^+ \geq 0$ and $d_i^- \cdot d_i^+ = 0$

Minimize

The *deviation function*:

Case A: Additive Multi-Attribute Utility Function

$$Z = 1 - E[U(\underline{X})] = \sum_{i=1}^{m} k_i (d_i^- + d_i^+)$$

Case B: Multiplicative Multi-Attribute Utility Function

$$Z = 1 - E[U(\underline{X})] = 1 - \frac{1}{k} \{(\prod_{i=1}^{m} [kk_i E[u_i(A_i(\underline{X}))] + 1]) - 1\}$$

$$= \frac{1}{k} \{(\prod_{i=1}^{m} [kk_i (1 - (d_i^- + d_i^+)) + 1]) - 1\}$$

Figure 2.23 – Mathematical Formulation of the Utility-Based Compromise Decision Support Problem (Seepersad 2001)

The formulation of the deviation function for the u-DSP involves four steps:

Step 1: Assess Utility Functions for Each Goal

Step 2: Combine Utility Functions for Individual Goals into a Multi-Attribute Utility Function

Step 3: Formulate System Goals

Step 4: Formulate the Deviation Function

Step 1: Assess Utility Functions for Each Goal

The formulation of the u-DSP begins with assigning a utility value for various values of goals. This utility value quantitatively reflects the preference of a designer. The determination of the utility value begins with the identification of the decision-maker's qualitative preference characteristics for each attribute (Fernández, Seepersad et al. 2001). Decision-maker preferences can be characterized as either monotonic or non-monotonic (Figure 2.24). Monotonic preferences describe instances in which a designer consistently prefers either strictly more (monotonically increasing) or less (monotonically decreasing) of an attribute. Non-monotonic preferences describe a scenario in which a designer has a preference for one specific value, and the closer a characteristic is to this ideal, the more it is desired.

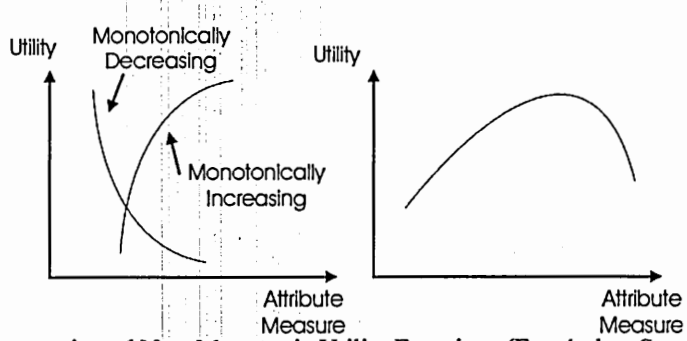


Figure 2.24 – Monotonic and Non-Monotonic Utility Functions (Fernández, Seepersad et al. 2001)

The preferences of the decision-maker are further qualified by assessing the curvature of his/her utility function. Concave utility functions imply a preference to act conservatively when faced with uncertainty (risk aversion). Convex utility functions imply a preference for risk (risk prone). These two curve types are presented in Figure 2.25. A risk-prone attitude means that a small change in the value of a goal from the most preferable value results in a large reduction in the utility. A risk-averse attitude means that a small change in the value of goal results in a relatively small change in the utility. Fernandez makes the assertion that most designers are risk averse under most circumstances; they prefer alternatives that offer on-target outcomes (Fernández, Seepersad et al. 2001).

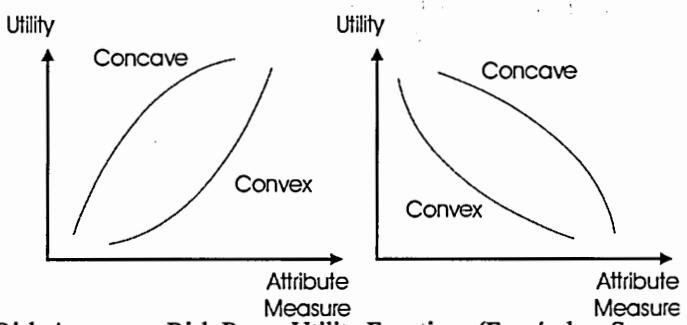


Figure 2.25 – Risk Averse vs. Risk Prone Utility Functions (Fernández, Seepersad et al. 2001)

Once the *qualitative* preference characteristics are established, the *quantitative* preference characteristics are determined. This stage of the decision-process involves the determination of the utility functions. These functions are determined by first having the designer to decide which value is ideal (a utility of 1), and which value is unacceptable (a utility of 0). Values in between these two extremes are assigned utilities via lotteries. A lottery is a hypothetical situation in which the outcome of a decision is uncertain and is used to assess a designer's preferences. (For more information, please consult a standard reference on utility theory, eg., (Keeney and Raiffa 1976).) Utility equations are then determined by fitting a curve to the points of designer preference (Figure 2.26).

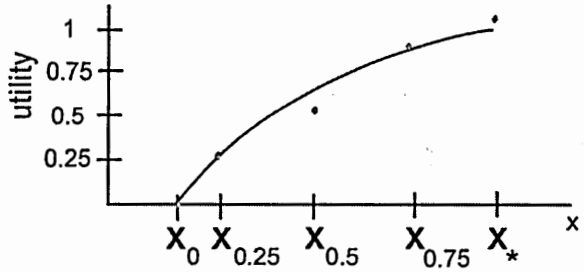


Figure 2.26 – Utility Curve Based on Designer Preferences (Fernández, Seepersad et al. 2001)

These concepts are further illustrated by returning to the cantilever beam tutorial example in Box 2.8.

Box 2.8 – Assessing Utility Functions for the Cantilever Beam Tutorial Example

Continuing from the discussion of the tutorial example in Box 2.7, it is now assumed that CustomTech has two conflicting objectives in their desire to manufacture custom cantliever beams. Instead of solely wishing to minimize cost, the manufacturer also wants to offer a product of high quality. This desire is reflected in the goal of minimizing the maximum displacement of the beam, w (Figure 2.27).

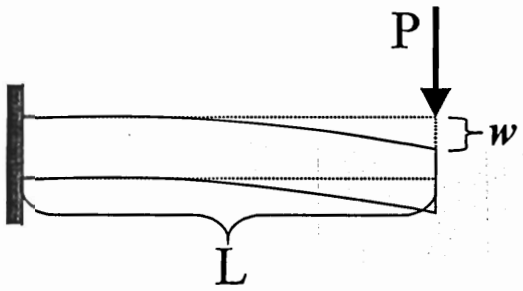


Figure 2.27 – The Deflection of a Cantilever Beam

The maximum deflection of a cantilever beam, for an end-loaded condition, is calculated by:

$$w_{\max} = \frac{-PL^3}{3EI} \quad \text{(mm)} \qquad [2.4]$$

where E is the modulus of elasticity (MPa) and I is the moment of inertia (mm^4).

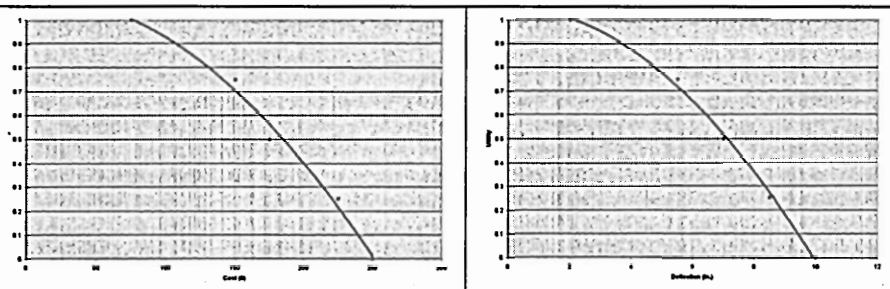
It is recalled from Box 2.1 that variety will be offered through the specification of the customers' desired beam length and maximum applied end load. It has been determined that the best way to minimize the maximum deflection is through increasing the modulus of elasticity and the moment of inertia of the beam. These properties can only be altered through the purchase of a new material, or through a re-design of the cross-section. The higher cost associated with materials and cross-sections of larger E and I create a tradeoff between the two goals of minimizing cost and minimizing deflection.

Two utility functions are created by using the methods discussed above. Assuming that the designer has monotonically decreasing, risk-averse preferences, the following utility values are assigned to specific variables via lottery questions:

Table 2.4 - Cantilever Beam Utility Function Assessment

Utility Value	Definition (Fernández et al., 2001)	Cost	Maximum Deflection
1	The decision-maker's ideal attribute level – beyond which the decision-maker is indifferent to further improvements in the attribute.	$75	2 in.
0.75	The decision-maker is indifferent between obtaining a design alternative with a 'desirable' attribute value for certain and a design alternative with a 50-50 chance of yielding either a tolerable or an ideal attribute level.	$150	5.5 in.
0.5	The decision-maker is indifferent between obtaining a design alternative with a 'tolerable' attribute value for certain and a design alternative with a 50-50 chance of yielding either an unacceptable attribute value or an ideal attribute value.	$175	7 in.
0.25	The decision-maker is indifferent between obtaining a design alternative with an 'undesirable' attribute value for certain and a design alternative with a 50-50 chance of yielding either a tolerable or an unacceptable attribute value.	$225	8.5 in.
0	The deicision-maker's unacceptable attribute level – beyond which he/she is unwilling to accept an alternative.	$250	10 in.

The resulting utility functions are plotted below in Figure 2.28.



(A) (B)

Figure 2.28 – Utility Curves for Cost (A) and Maximum Deflection (B)

As can be seen in the above figure, polynomial curves are fitted to the plotted points. The resulting independent utility equations are then:

$$u_{cost} = -0.00002C^2 - 0.0002C + 1.1167 \qquad [2.5]$$

$$u_{deflection} = -0.0101w^2 - 0.0069w + 1.0609 \qquad [2.6]$$

The next step in this decision process is combining these two utility functions into a multi-attribute function.

Step 2: Combine Utility Functions for Individual Goals into a Multi-Attribute Utility Function

After individual utility functions are developed, they are combined into a multi-attribute utility function. The method in which way this is done is dependent on the relevant independence assumptions made by the designer. The types of independence are detailed in Keeny and Raiffa (Keeney and Raiffa 1976).

The next stage of this step is the assessment of the scaling constants for the multi-attribute utility function. The scaling constants reflect the preference between attributes. They can be methodically determined by solving a system of equations where different attributes are compared in order to evaluate the scaling constants.

The end result of these stages is the multi-attribute utility function. An example of this function, when under the assumptions of utility and additive independence (for example), is presented in the following equation:

$$U = \sum_{i=1}^{n} k_i u_i(A_i) \qquad [2.7]$$

where $u_i(A_i)$ is an individual utility function of an attribute (A_i), k_i is a scaling constant, and U is the total expected utility. After this is developed, the consistency of the expressed preferences is double checked to ensure that none of them conflict.

Box 2.9 – Formulating a Multi-Attribute Utility Function for the Cantilever Beam Tutorial Problem

Returning to the cantilever beam example, it is assumed that there is no interaction between a designer's preferences for the different attributes. As a result, the designer's preferences can be modeled with the additive multi-attribute utility function assumptions. The resulting expected utility function is:

$$U = k_{cost} u_{cost} + k_{deflection} u_{deflection} \qquad [2.8]$$

The scaling constants are determined by solving the following set of equations:

$$k_{cost} u(C_0) + k_{deflection} u(w_0) = k_{cost} u(C_q) + k_{deflection} u(w_1) \qquad [2.9]$$

$$k_{cost} u(C_0) + k_{deflection} u(w_0) = k_{cost} u(C_1) + k_{deflection} u(w_q) \qquad [2.10]$$

where the subscript 0 represents the value of the attribute at a utility of 0, the subscript 1 represents the attribute's ideal value, and the subscript q represents the value of the attribute at which a designer is indifferent between the two alternatives (C_0, w_0) and (C_1, w_q).

Step 3: Formulate System Goals

The system goal, as previously discussed, is for the utility of the system to reach the ideal value (= 1) (Seepersad 2001). As such, the system goal is expressed as:

$$E[u_i(A_i(X))] + d_i^- - d_i^+ = 1 \qquad [2.11]$$

Box 2.10 – Formulating Sytem Goals for the Cantilever Beam Tutorial Problem

For the cantilever beam running example, there are two system goals, the cost utility function reaching the ideal utility,

$$E[u(C)] + d_C^- - d_C^+ = 1 \qquad [2.12]$$

and the deflection utility function reaching the ideal utility:

$$E[u(w)] + d_w^- - d_w^+ = 1 \qquad [2.13]$$

Step 4: Formulate the Deviation Function

The deviation function is formulated to minimize the deviation from the target utility (i.e., 1) which is mathematically equivalent to maximizing the utility. The additive multi-attribute utility function is provided below (Seepersad 2001):

$$Z = 1 - E[U(X)] = \sum_{i=1}^{m} k_i (d_i^- + d_i^+) \qquad [2.14]$$

Box 2.11 – Formulation of the Deviation Function for the Cantilever Beam Example

The final step of the cantilever beam example problem is the formulation of the deviation function. The resulting equation, Equation 2.15, is based on the results obtained from Equation 2.8:

$$Z = 1 - E[U(X)] = k_C(d_C^- + d_C^+) + k_w(d_w^- + d_w^+) \qquad [2.15]$$

2.2.4 Critical Evaluation of the Utility-Based Compromise Decision Support Problem

The Utility-Based Compromise Decision Support Problem provides a decision construct in which a designer can model multiple, conflicting objectives. It quantitatively models a designer's preference in the conditions of uncertainty. It has been used to design product platforms (Seepersad 2001) – specifically a family of absorption chillers, and a family of cordless drills. It provides a construct in which a designer can search for "satisficing solutions" (Simon 1996) – solutions that are satisfactory, or "good enough" – instead of the optimal solution that may be impossible due to practical limitations.

The Utility-Based Compromise Decision Support Problem is the tool that is used to augment the PPCTM in order for it to empower designers with the ability to handle the multiple objectives (and the inherent tradeoffs) frequently found in the development of product and process platforms.

2.3 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD

The role of Chapter 2 in this thesis is two-fold: the presentation, discussion and critical analysis of (1) the Product Platform Constructal Theory Method, and (2) the constructs that will be used to augment this method to enable it to cope with multiple

objectives in the development of both product and process platforms in the presence of non-uniform demand.

As presented in Section 1.4.1, theoretical structural validity is established through the searching and referencing of literature related to each construct employed in the design method. Theoretical structural validity for Hypotheses 1, 2, and 3 is established in this chapter through the discussion of the advantages, limitations, and accepted domains of the constructs of the PPCTM and the u-cDSP, as illustrated in Figure 2.29.

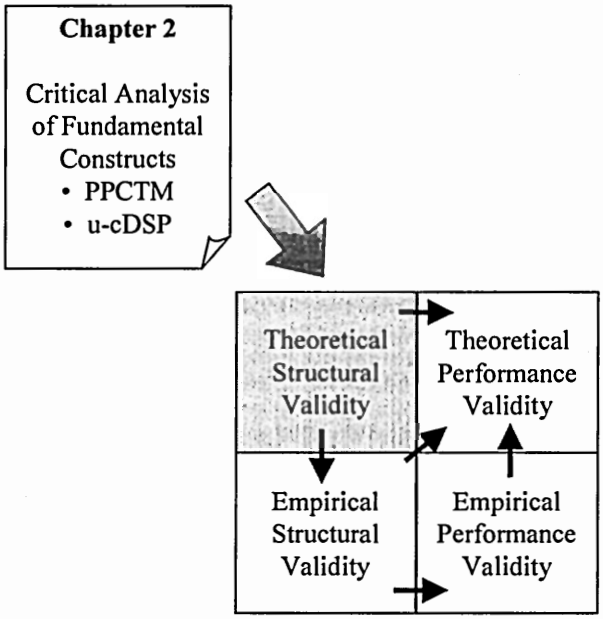


Figure 2.29 – Theoretical Validation Established in Chapter 2

In Section 2.1, the Product Platform Constructal Theory Method is presented in detail. Its theoretical foundations in hierarchic systems theory and constructal theory are discussed. A description of the link between constructal theory and Hernandez's abstraction into the realm of product family design is presented in Section 2.1.1.

The six steps of the PPCTM are detailed in Section 2.1.2. The result of the application of the PPCTM for product platform design is the synthesis and hierarchical organization of multiple modes for managing product variety. Specifically, by posing the

platform design as a optimal access problem, the ranges of applciation of multiple modes are chosen through formulating and solving the resulting multistage decision problem as a dynamic program.

In Section 2.1.3 a critical analysis of the PPCTM reveals four fundamental limitations: an inability to deal with multiple objectives, a focus on optimization, a working assumption of uniform demand across the market (the fundamental focus of this thesis), and a lack of consideration for the manufacturing process of the customized product. The augmentation of the PPCTM to alleviate these specific limitations is the core focus in this research.

The limitations regarding multiple objectives and the focus on optimization require the description of a specific theoretical construct. The utility-based compromise Decision Support Problem (u-cDSP), along with its two theoretical constructs (the compromise Decision Support Problem and utility theory) is presented in Section 2.2 as an answer to these limitations of the PPCTM. The u-cDSP provides structure and support for engineering decisions involving multiple attributes. It also provides a mathematically rigorous and axiomatic method for quantitatively capturing a designer's preferences. The four-step formulation of the u-cDSP guides a designer through a methodical manner of capturing his or her preferences in a multi-attribute utility function. Methods for the checking the consistency of these functions are included in the formulation of the u-cDSP, which further strengthens theoretical structural validity of this augmentation.

In Chapter 1, the motivation for developing a methodical approach to the design of product and process platforms for customizable products in a market of non-uniform demand is established. In this chapter, a review of the theoretical foundation to be used

for the augmentation of the PPCTM is presented. The augmentation of the Product Platform Constructal Theory Method with the utility-based compromise Decision Support Problem is presented in Chapter 3. The application of the augmented method is applied to two example problems in Chapters 4 and 5.

CHAPTER III

A METHOD FOR DESIGNING PRODUCT AND PROCESS PLATFORMS AS A PROBLEM OF ACCESS IN A GEOMETRIC SPACE OF NON-UNIFORM DEMAND

The primary objective embodied in this thesis is the augmentation of the Product Platform Constructal Theory Method (PPCTM). The focus of this augmentation is to properly extend the formulation and the approach of the PPCTM to be able to handle the design of product platforms of customizable products in market spaces of non-uniform demand. This focus is captured in the primary research question (Section 1.3):

> *How can the Product Platform Constructal Theory Method be augmented in order to cope with the development of product platforms in a market of non-uniform demand?*

The accomplishment of this primary focus depends on the success of additional related research tasks, represented in Section 2.1.3 as Research Questions 2 and 3. The desire to be able to handle a market space of non-uniform demand is coupled with a need for a multi-stage decision process that can handle the multiple objectives and inherent tradeoffs typically encountered in platform design (Research Question 2, Section 1.3). In order to fully capture the design of a customizable product in markets of non-uniform demand, one must also consider the design of the manufacturing process for the line of products. As such, an opportunity for research in the application of the PPCTM to the domain of production process design is presented (Research Question 3, Section 1.3). These three major research thrusts (non-uniform demand, multi-objective decision

making, and process design for customized products) are the focus in the augmentation of the PPCTM.

In this the third chapter, the augmented PPCTM is presented. An overview of several non-uniform demand modeling techniques are presented in Section 3.1. Several concepts necessary for the application of the PPCTM to process design are presented in Section 3.2. In Section 3.3, details of how the u-cDSP is infused into the PPCTM are discussed. Finally, in Section 3.4, the augmented PPCTM is presented. Similar to the presentation of the previous chapters, the customizable cantilever beam tutorial problem is used in order to assist in the explanation of difficult concepts.

The effort to establish theoretical structural validity is continued in this chapter by verifying the internal consistency of infusing the proposed augmentations into the PPCTM. A critical evaluation of the resulting augmentation is also provided at the closure of this chapter. As shown in Figure 3.1, the role of this third chapter is the presentation of the augmented PPCTM in order to provide theoretical structural validation.

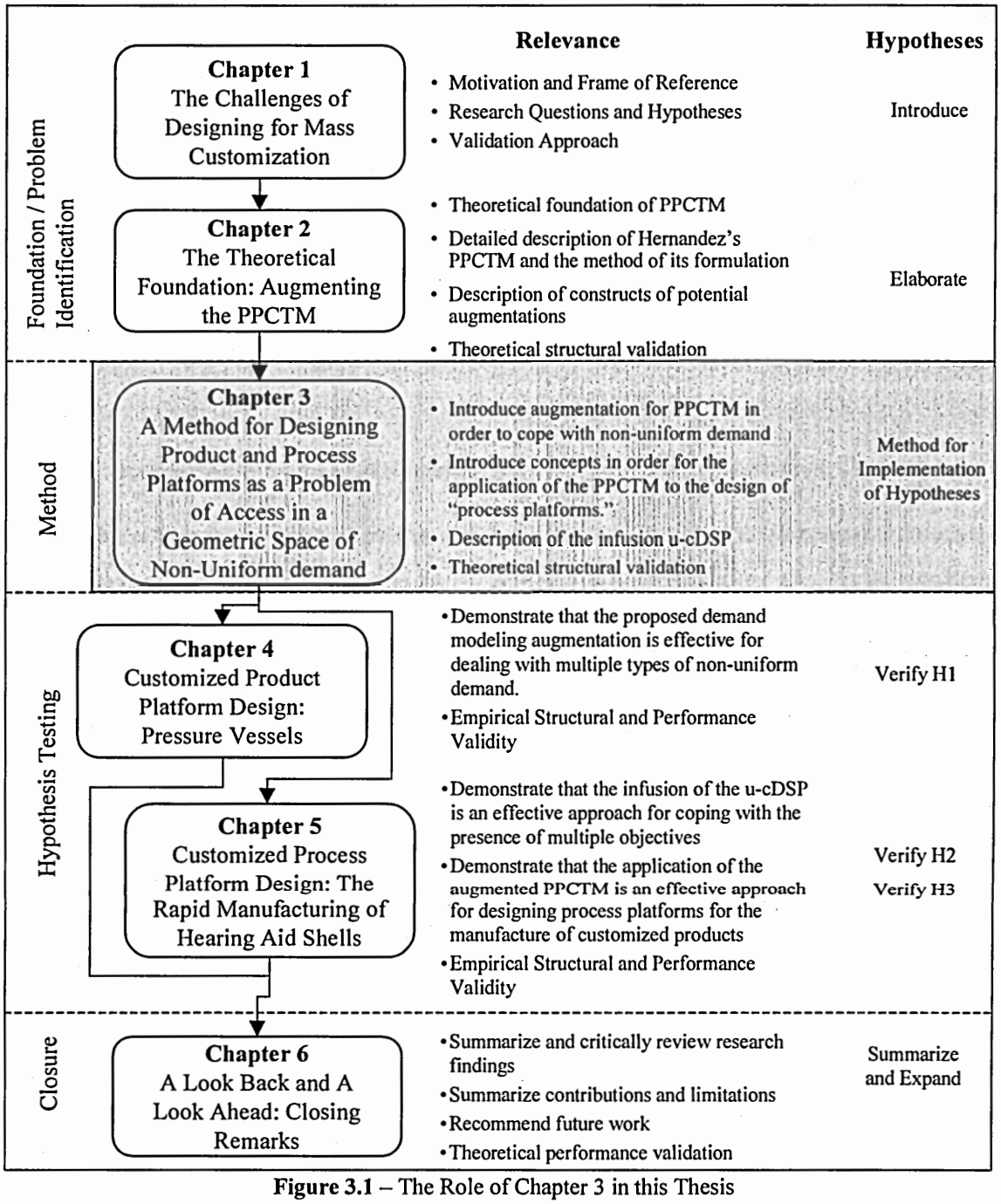


Figure 3.1 – The Role of Chapter 3 in this Thesis

3.1 THE MODELING OF NON-UNIFORM DEMAND

In the PPCTM, demand is assumed to be uniform across the entire market space. This approximation does not adequately capture the complexity of a traditional market space for a customized product (Section 1.3). In order to augment the PPCTM to cope with non-uniform demand, demand-modeling techniques must first be established.

While there are many different methods for modeling demand established in the literature, each method is focused on *forecasting* demand. Typically oriented towards the use of different regression techniques to fit curve approximations to previous levels of demand (Hopp et al., 2001), these modeling techniques are not applicable to this research. Currently, a designer developing product platforms with the PPCTM must assume that the market is fixed; i.e., the manufacturing enterprise will not vary the span of products that will be offered to the customer. The ability to handle the design of a product platform for an expanding market is beyond the scope of this research and is suggested as future work (see Section 6.4).

Since the PPCTM can only be used to handle a "snapshot" of the current market (or at most, an average of the market trend), mathematical methods for modeling demand are presented in this thesis as a means of handling markets of non-uniform demand. For the augmentation of the PPCTM, it is desired to provide a designer with the ability to design a product platform for any type of demand scenario. In the remainder of this section, three different suggested methods of modeling demand for the application of the PPCTM are presented:

- as a discrete function,
- as a linear continuous function, and

- as a normal distribution.

The formulation of the objective function is the primary step that is affected through the inclusion of a non-uniform demand assumption in the PPCTM (as outlined in Step 3, Section 2.1.2). For each modeling technique, a representation of a typical formulation of an objective function is presented for both a discretized and a continuous analysis of the market space.

3.1.1 Discrete Function

One way to model a non-uniform demand scenario is to define a series of spaces within the market space as having uniform demand. This is accomplished by defining a set of bounds for a series of sub-spaces with different values of demand within the overall market space.

An objective function for an *n*-dimensional market space with non-uniform demand analyzed as a discrete function is formulated as:

$$O = \sum_{i=1}^{m} \frac{1}{D_i} \int_{n_i}^{n_{i+1}} \cdots \int_{y_i}^{y_{i+1}} \int_{x_i}^{x_{i+1}} O(x, y, \ldots n)\, dx dy \ldots dn \qquad [3.1]$$

for a continuous analysis of the market space, and as:

$$O = \sum_{i}^{n} \frac{1}{D_i} \sum_{i=y_{i,\min}}^{y_{i,\max}} \sum_{j=x_{i,\min}}^{x_{i,\max}} \cdots \sum_{k=n_{i,\min}}^{n_{i,\max}} O(x_j, y_i, \ldots n_k) \qquad [3.2]$$

for a discretized analysis of the market space.

In Equations 3.1 and 3.2, O is the objective value and m is the total number of discrete divisions modeled in the space. The subscripts $i+1$ and i pertain to the values of the design parameters that are the bounds of each division. The *max* and *min* subscripts refer to the upper and lower bounds of each dimension, respectively. This is further illustrated by returning to the cantilever beam tutorial example in Box 3.1.

Box 3.1 - Modeling Demand of the Cantilever Beam Tutorial Example as a Discrete Function

Consider a market space for the cantilever beam example where there is a significant difference in demand among product variants. Specifically, the requirements of customers of customizable cantilever beams seem to aggregate near the bounds of the loading specification, as shown in Figure 3.2.

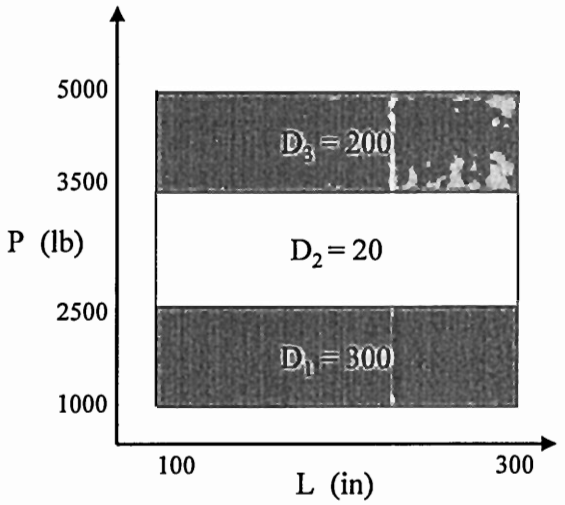


Figure 3.2 – Discrete Demand Scenario for Cantilever Beam Example

The formulation of the objective function (in a discrete analysis) is shown in Equation 3.3.

$$O = \frac{1}{D_1}\sum_{i=1000}^{2500}\sum_{j=100}^{300} O(L_j,P_i) + \frac{1}{D_2}\sum_{i=2501}^{3500}\sum_{j=100}^{300} O(L_j,P_i) + \frac{1}{D_3}\sum_{i=3501}^{5000}\sum_{j=100}^{300} O(L_j,P_i) \qquad [3.3]$$

where D_1, D_2, and D_3 represent the demand (number of products per year) presented in Figure 3.2.

This method of modeling demand is appropriate for market spaces where some products are significantly more requested than the others. This modeling method is especially useful when portions of the market space have products that have *no demand*, as it allows a designer to model this discontinuity accurately.

3.1.2 Linear Continuous Function

Another manner in which one can model demand is through a simple linear approximation. This is an appropriate method to estimate a market space where customers tend to prefer products on the upper (or lower) end of the market space. This technique also provides a manner in which demand can be expressed as a function of the design parameters.

A typical objective function, for an *n*-dimensional space, is shown in Equations 3.4 and 3.5.

$$O = \int_{n_{\min}}^{n_{\max}} \cdots \int_{y_{\min}}^{y_{\max}} \int_{x_{\min}}^{x_{\max}} \frac{1}{D(x, y, \ldots n)} O(x, y, \ldots n) dx dy \ldots dn \qquad [3.4]$$

$$O = \sum_{i=y_{\min}}^{y_{\max}} \sum_{j=x_{\min}}^{x_{\max}} \cdots \sum_{k=n_{\min}}^{n_{\max}} \frac{1}{D(x_j, y_i, \ldots n_k)} O(x_j, y_i, \ldots n_k) \qquad [3.5]$$

where $D(x,y)$ would be of the form:

$$D(x, y, \ldots n) = ax + by + \ldots + cn + d \qquad [3.6]$$

where a, b, c, and d are appropriate coefficients for the approximation of the linear trend. The application of this demand modeling technique in the cantilever beam example is presented in Box 3.2.

Box 3.2 - Modeling Demand of Cantilever Beam Tutorial Example as a Linear Function

It is now assumed that the customers of CustomTech tend to prefer cantilever beams that are longer and can withstand higher end-loads. The resulting linear non-uniform demand model of the market space is shown in Figure 3.3.

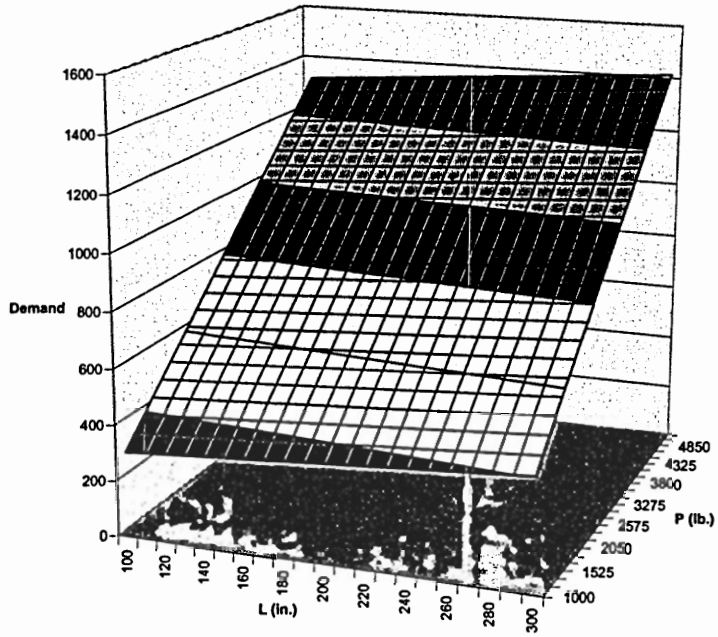


Figure 3.3 – Linear Demand Scenario for Cantilever Beam Example

The formulation of the objective function (in a discrete analysis) is shown in Equation 3.7:

$$O = \sum_{j=1000}^{5000} \sum_{i=100}^{300} \frac{1}{\left(0.5L_i + 0.25P_j + 5\right)} O(L_i, P_j) \qquad [3.7]$$

3.1.3 n-Dimensional Gaussian Distribution

The third manner in which demand is modeled in this thesis is with the use of a Gaussian (normal) distribution function. The normal distribution is an effective manner of modeling the probability of a product being demanded. A one-dimensional Gaussian distribution follows the form (Hayter, 2002):

$$y = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \qquad [3.8]$$

where μ is the mean, and σ is the standard deviation (equal to the square root of the variance). The mean identifies where the peak of the density of the demand occurs, and the standard deviation indicates the girth of the bell curve of the normal distribution.

For the infusion of this modeling technique into the PPCTM, its formulation must be general enough to be able to model a normal distribution for every dimension of the market space. This is done by taking the product of each one-dimensional Gaussian function for each dimension in the market (i.e, an n-dimensional space requires a normal distribution function that is the product of n one-dimensional functions) (Hayter, 2002). The resulting function approximates demand as a function of each dimension in the space. The resulting objective functions are identical to that which is presented in Equations 3.4 and 3.5 but with the demand evaluated by the equation:

$$D(x,y) = \prod_{i}^{n} \left(A_i \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{i-\mu_i}{\sigma_i}\right)^2} \right) \qquad [3.9]$$

for an n-dimensional market space. The subscript, i, denotes the statistical properties respective to each individual dimension. The variable A is a scaling coefficient that translates the *probability* of a product being ordered to the *number of products* being ordered. This concept is applied to the cantilever beam tutorial example in Box 3.3.

Box 3.3 – Modeling Demand of the Cantilever Beam Tutorial Example as a Gaussian Distribution

It is now assumed that the orders from the customers of CustomTech tend to fall within a pattern of normal distribution. After an extensive survey of the past ordering periods, the manufacturers at CustomTech estimate the following statistical properties for the demand of their product:

Table 3.1 – Cantilever Beam Normal Distribution Parameters

μ_L	180 in.
μ_P	3200 lbs.
σ_L	27
σ_P	387
A	1600 beams

The resulting demand in the customized space is visualized in Figure 3.4.

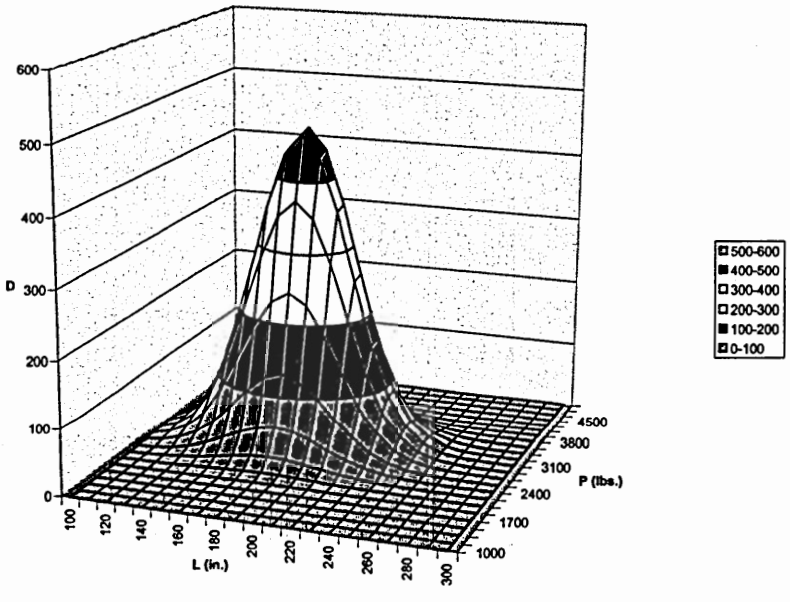


Figure 3.4 – Gaussian Demand Scenario for Cantilever Beam Example

The objective function to be formulated for the PPCTM is calculated as:

$$O = \sum_{j=1000}^{5000} \sum_{i=100}^{300} \left[1 \Big/ A \left(\frac{1}{\sigma_{L_i}\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{L_i-\mu_{L_i}}{\sigma_{L_i}}\right)^2} \right) \left(\frac{1}{\sigma_{P_j}\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{P_j-\mu_{P_j}}{\sigma_{P_j}}\right)^2} \right) \right] O(L_i, P_j) \qquad [3.10]$$

These three methods for modeling the demand of a market space represent a small number of the possible ways of capturing demand information. Conceptually, there are an infinite number of modeling techniques available. These three methods are presented here because they represent a logical, effective, and efficient way of modeling demand.

These modeling techniques are infused into the first step of the PPCTM (see Section 3.4.1) where a designer models the market space as a geometric space. It is asserted that the successful infusion of these modeling techniques into the PPCTM will provide empirical structural validity that the PPCTM can handle any type of non-uniform demand (see Chapter 4).

3.2 ABSTRACTING THE PRODUCT PLATFORM CONSTRUCAL THEORY METHOD TO PROCESS DESIGN FOR CUSTOMIZED PRODUCTS

The core issue of transitioning to mass customization is how to design both the product and the manufacturing production process to be able to offer affordable variety (Section 1.1.2). While the Product Platform Constructal Theory Method is currently formulated for the design of product platforms for customizable products, it does not provide a framework in which the production process can be concurrently designed.

It is the author's assertion that an abstracted form of the PPCTM can be applied to the design of a production process for customized products (Research Hypothesis #3, Section 1.3). In this section, several concepts are defined for the successful application of the PPCTM to this new domain. Specifically, the realization of a line of process parameters for a particular market is conceptually abstracted of as a problem of optimal access, as done similarly for product platform design in the original formulation of the PPCTM (Section 2.1.1).

3.2.1 Definition of Process Platform

The crux of the design of a production process is the determination of values of process parameters that result in the best value for an objective. This is a very complex problem, as a designer must select a bevy of process parameters while keeping multiple objectives in mind. Examples of process parameters to be selected include batch size, speed of operation, the number of machines to use, and the type of machines to use. Common manufacturing objectives include minimization of machine usage time, minimization of lead-time, maximization of throughput, and the maximization of product quality.

Each of these objectives is secondary to the overall objective of being able to meet production capacity needs (i.e., the process must be able to produce the number of products demanded within the amount of time promised to the customer). The imposed constraints of cycle time and manufacturing capacity are therefore the key drivers for the design of a manufacturing process. As such, the design of the production process is directly related to the amount of throughput required (i.e., the values of process parameters are determined so that for a given capacity requirement, the production process is as effective as possible).

When confronted with the task of designing a production process for a mass customized product, designers are confronted with two separate dilemmas:

i) How can the production process be designed in order to handle the fluctuating demand of the product?

ii) How can the production process be designed in such a manner that multiple products can be manufactured affordably?

It is the author's assertion that the answer to both of these quandaries lies in the efficiency of the design process itself. Specifically, it is inefficient for the production process to be redesigned for each new capacity requirement. As outlined in Section 1.1.2, the success of mass customization relies on the flexibility and agility of the manufacturing process. Therefore, the design of the manufacturing process itself should also be flexible. To satisfy this need for efficient design of a manufacturing process, the concept of a *process platform* is introduced.

> *A **process platform** is defined as a set of common process parameters from which a stream of derivative parameters can generate a customized product efficiently for changes in capacity requirement.*

It is recalled from Section 1.2 that a product platform is defined as a set of common components, modules, or parts from which a stream of derivative products can be created (Lehnerd, 1987). The concept of a process platform is very similar: process parameters are commonalized to satisfy a range of capacity requirements. Just as the commonality of design parameters in a product platform lowers the cost of offering of product variety, the commonality of process parameters in a manufacturing process platform lowers the cost of the setups encountered with reconfiguring a production workstation for different capacity requirements. This concept of a process platform is illustrated through its application to the cantilever beam tutorial example in Box 3.4.

In previous applications of the cantilever beam example in this thesis, the context of the problem has been focused on the design of a product platform. For the purpose of illustrating concepts associated with process platform design, the focus of the example is shifted toward the design of the manufacturing process.

Box 3.4 – The Process Platform Problem of the Cantilever Beam Tutorial Example

The customizable cantilever beam manufacturing system is composed of a single process: cutting the beams to the required length from the raw length of material. This process has a series of parameters that must be appropriately determined: the speed at which the material is cut, the size of each production batch, the number of machines, and the selection of the cutting machine (some machines can cut different amounts, and/or at different speeds, than others). There exists a constraint on the manufacturing process such that a beam must be completed within a week after the placement of its order.

Due to the volatile nature of the demand of the customizable beams, the manufacturer recognizes the value of developing a process platform – it is too costly to reconfigure and re-determine the values of the process parameters due to a simple change in demand.

There are two objectives in the development of this production process platform: the minimization of cost, and the maximization of throughput. There exists an inherent tradeoff between these two objectives: the cost associated with purchasing faster machines and multiple machines conflict with the desire for higher throughput. The manufacturer is therefore in need of a methodical manner in which to develop an appropriate process platform.

3.2.2 Definition of Space of Capacity

The core of the Product Platform Constructal Theory Method is the abstraction of product platform design as a problem of optimal access in a geometric space (Section 2.1). In order to successfully apply the constructs of the PPCTM to the design of a production process platform, concepts similar to those of the PPCTM must be developed.

The first step of the PPCTM is the definition of the market space (Section 2.1.2). For product platform design, this geometric space is defined by the determination of the design specifications that will be varied in order to offer a customized product. The number of design specifications determines the dimension of the geometric space; the range of variety that will be offered for each parameter determines its bounds.

Similar to the design of products using the PPCTM, the geometric space for process platform design is dependent on the parameter chosen to be varied. The focus in process platform design is to offer variety in manufacturing capacity affordably through the commonalization of process parameters (Section 3.2.1). Since the production capacity of the manufacturing system is the only parameter being varied through the development of a process platform, it serves as the lone dimension of the resulting geometric space. This geometric space is defined as the space of capacity:

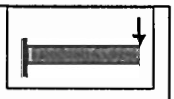
> *A **space of capacity** is the range of production capacity that a manufacturing enterprise is willing to satisfy with a process platform.*

The bounds of this single dimension are determined by the manufacturing enterprise as the amount of production capacity that their manufacturing system should satisfy.

With this geometric space defined, the crux of the application of the PPCTM for production process platform design is similar to that of its use for product platform

design: the synthesis of multiple modes of managing variety in order to provide any variant within the geometric space (i.e., for product design each variant in the space is a different product; for designing processes each variant in the space represents a different production capacity requirement). The appropriate definition of a mode of managing variety for process platform design is provided in the following section. The cantilever beam tutorial example is continued in the context of manufacturing process design in Box 3.5.

Box 3.5 – Space of Capacity for the Cantilever Beam Tutorial Example



For the successful application of the PPCTM to this example, an appropriate space of customization must be determined. The definition of this space relies solely on the definition of the range of capacity variety that the manufacturing enterprise wishes to offer. After considering the history of the amount of beams demanded, and performing a demand forecasting analysis, CustomTech decides that the manufacturing system typically faces production capacity requirements between 100 to 1000 beams per week. The resultant space of capacity for the customizable cantilever beam manufacturing system is presented in Figure 3.5.

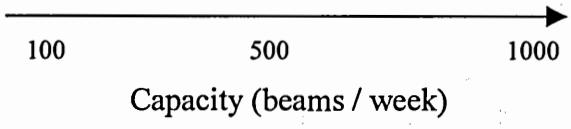


Figure 3.5 – The Space of Capacity for the Cantilever Beam Example

3.2.3 Definition of Modes for Managing Process Variety

As discussed in Section 2.1.1, the focus in the PPCTM is the organization and synthesis of modes of managing product variety – any generic approach in product design or its manufacturing process for achieving product customization systematically (Hernandez, 2001). Common modes for managing product customization include modular design, platform design, robust design, dimensional customization, and adjustable customization. These modes serve as the linking mechanism between the individual products that compose a product family.

To successfully apply the PPCTM in the realm of designing production processes, a similar construct must be developed. As such, the term "modes for managing process variety" is introduced and defined below.

> *A **mode of managing process variety** is any generic approach in the design of a production process for achieving a change in manufacturing capacity.*

Examples of modes of managing process variety include, but are not limited to, process parameter standardization, batch size commonalization, machine type commonalization, and the modular combination of machine capacity (i.e., adding a new machine to the manufacturing system). These modes serve as the linking mechanism between the individual production capacity requirements that compose a process family. This concept is explored through further development of the manufacturing process design of the cantilever beam tutorial.

Box 3.6 – Modes of Managing Process Variety in the Cantilever Beam Tutorial Problem

For the application of the PPCTM to the design of the cantilever beam production process, the manufacturer must determine appropriate modes of managing process variety. The selection of these modes is based solely on brainstorming. Typically, these modes for managing variety involve the standardization of a process parameter that has direct influence on the production capacity and is easily controlled by the manufacturing engineer. Similar to the selection of modes for product design, technical and economical limitations exist in the proper selection of modes of managing process variety.

For the development of a process platform for the cantilever beam example, the manufacture has chosen the following modes:

- *Standardization of cutting rate*: The rate at which the beams are cut has a huge influence on the production rate of the manufacturing process. High fidelity changes in production capacity can be made through control of the cutting rate.
- *Standardization of machine type*: There are several types of machines available for the production process. Different machine types offer variety in batch sizes and production rate. This mode offers large, discrete changes in production capacity.
- *Modular combination of machines*: Adding another machine to the manufacturing system provides a very large increase in production capacity. While much greater production capacity can be achieved through the implementation of this mode, the manufacturer must take into account the significant increase in cost.

With a means for linking the geometric space established, a systematic method for their synthesis and the determination of their appropriate range of application is presented in the next section.

3.2.4 Design of Process Platforms for Customizable Products as an Access Problem

With these definitions established, a more formal explanation of how the PCPTM can be applied to process design is provided in this section. While the specific details of the formulation of the PPCTM are presented in Section 3.4, a general overview of the process is provided here.

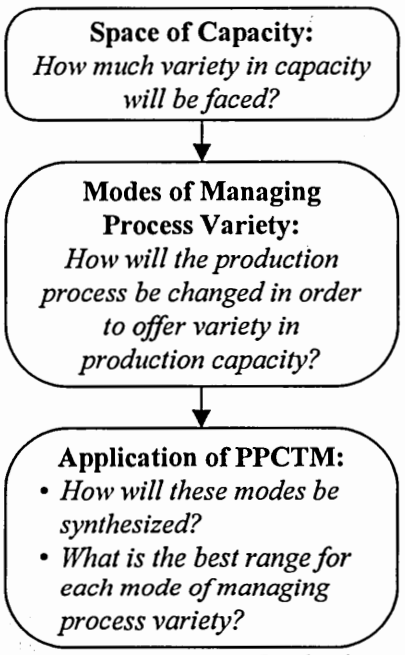


Figure 3.6 – Applying PPCTM Concepts to Production Process Design

As can be seen in Figure 3.6, the design of a process platform with the PPCTM begins with defining the range of production capacity that a manufacturing enterprise is willing to offer. This range of capacity is directly connected to the expected fluctuating demand of the product, thus there exists a relation to the need to consider non-uniform demand for the design of a customized product.

Once the range of production capacity to be offered is determined, the modes for managing process variety are identified. Each mode should provide an affordable and feasible manner in which the manufacturing system's capacity can be altered. Similar to the use of the PPCTM for product design, these modes are ordered in a hierarchic fashion, placing those modes that offer higher fidelity changes in the manufacturing system's capacity at the bottom of the hierarchy.

The determination of the range of each mode of managing process variety serves as the baseline decision for each decision stage of the PPCTM. These decisions are based on each mode's influence on the objective to be improved through the application of the PPCTM. These decisions must be formulated in a systematic manner in order to capture the nature of the multistage decision process and the presence of multiple, conflicting objectives.

The end result of the application of the PPCTM to the design of a production process is the synthesis, hierarchic organization, and determination of the ranges of multiple modes of managing process variety in order to construct a process platform. A final illustration of the cantilever beam tutorial problem in the context of the design of its production process is provided in Box 3.7.

Box 3.7 – Applying PPCTM to the Process Platform Problem of the Cantilever Beam Tutorial Problem

With the fundamental process design constructs defined, the manufacturer of the customizable cantilever beams is ready to apply the PPCTM.

The manufacturer must first appropriately order the modes of managing process customization, taking care to place those with the ability to offer high fidelity changes in

production capacity near the bottom of the hierarchy. The resulting hierarchy is shown in Figure 3.7.

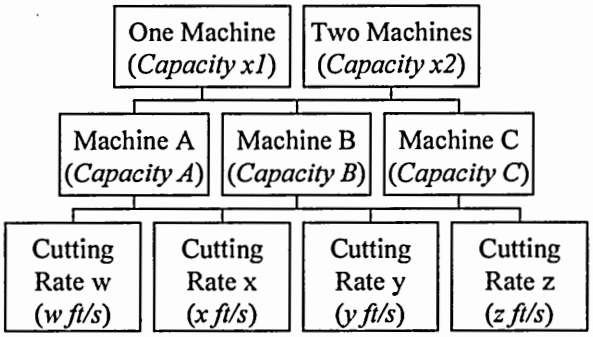


Figure 3.7 – Synthesis of Modes of Managing Process Variety for the Cantilever Beam Example

A multi-stage decision process that can handle the conflicting objectives of minimization of cost and maximization of throughput is then implemented and solved in order to select the appropriate range of commonalization of each mode. The end result of this process is a process platform that provides the best solution for both objectives - the selection of the range of commonalization of each mode in a hierarchical arrangement. A graphical representation of a solution for this problem is provided in Figure 3.8.

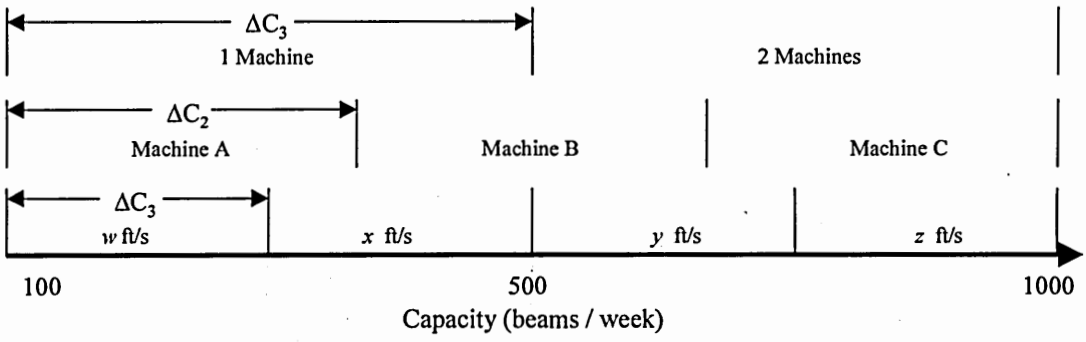


Figure 3.8 – Graphical Representation of the Process Platform Design of the Cantilever Beam Example

From this solution, the manufacturer at CustomTech knows which process parameter values should be chosen for each production capacity requirement. For example, if the capacity requirement for the week is determined to be 550 beams, the manufacturer knows that two "Machine Bs" should be used at a cutting rate of *y* ft/s.

3.3 COPING WITH MULTIPLE OBJECTIVES WITH THE UTILITY-BASED COMPROMISE DECISION SUPPORT PROBLEM

As discussed in Section 2.1.3, one of the limitations of the use of the PPCTM is an inability to handle multiple objectives in its decision-making stages. In Section 2.2, the utility-based compromise Decision Support Problem (u-cDSP) is presented as a suitable decision construct to satisfy this limitation in the PPCTM. In this section the manner in which the u-cDSP is infused into the PPCTM is discussed.

In order to illustrate this infusion, the specific steps of the PPCTM that are influenced by the constructs of the u-cDSP are presented in Section 3.3.1. Further clarification is offered through the use of the cantilever beam tutorial problem. A critical analysis of this infusion is presented in Section 3.3.2.

3.3.1 The Infusion of the u-cDSP into the PPCTM

In the implementation of the PPCTM, a designer is required to follow six-steps, as shown in Figure 3.9 (refer to Section 2.1.2 for more detail).

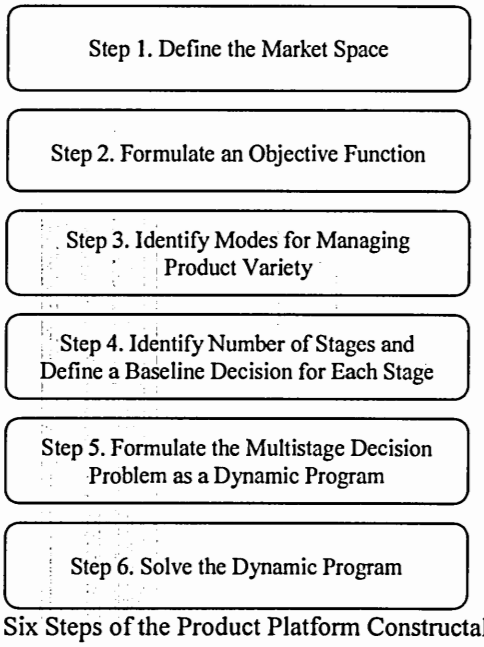


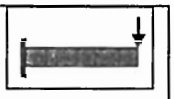
Figure 3.9 – The Six Steps of the Product Platform Constructal Theory Method

Two steps, Step 2 and Step 4, deal with the development and formulation of the multistage decision problem. As such, these steps are the focus of the infusion of the u-cDSP.

PPCTM Step 2: Formulate an Objective Function

The second step of the PPCTM involves the identification of one parameter to be improved by the method (Section 2.1.2). Through the infusion of the u-cDSP into the PPCTM, this limitation no longer exists. A designer is free to identify all of the objectives that are to be improved through the use of the PPCTM. As a result, the problem of designing a product or a process platform is modeled more accurately. An illustration of this change in the PPCTM is illustrated in Box 3.8.

Box 3.8 – Formulation of Objective Functions for the Cantilever Beam Tutorial Problem



It is recalled that the objectives of CustomTech in the development of the customizable cantilever beam are to minimize cost and to minimize maximum deflection. The description of the formulation of these objective functions has been previously presented in Boxes 2.2 and 2.8.

There are inherent tradeoffs in these two objectives. A designer cannot simultaneously minimize cost and minimize minimum deflection - materials with a larger modulus of elasticity or a larger moment of inertia are much more costly. Through the identification of these two objectives, the designers are ready to move onto the next step of the PPCTM.

PPCTM Step 4: Identify Number of Stages and Define a Baseline Decision for each Stage

In the fourth step of the PPCTM, a designer creates an order of application for the modes of managing product customization. Once the order is established, a designer then identifies the number of stages and defines a baseline decision for each stage. With the addition of multiple objectives, a designer is in need of a manner to combine all of the objective functions developed in Step 2 for efficacious analysis. The infusion of the u-cDSP alleviates this problem.

Specifically, the four steps of formulating a utility-based compromise Decision Support Problem (Section 2.2.3) are implemented in this step, as shown in Figure 3.10.

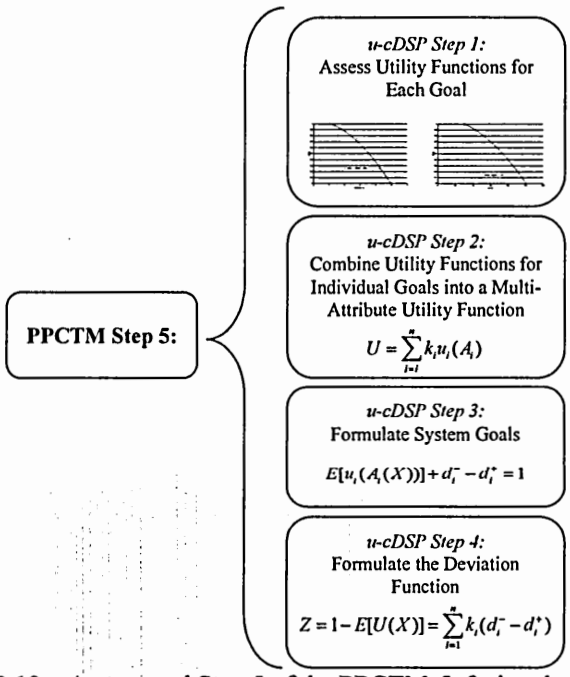


Figure 3.10 – Augmented Step 5 of the PPCTM: Infusing the u-cDSP

The result of this augmentation is the formulation of a utility-based compromise Decision Support Problem for each decision stage of the problem. The goal and deviation functions formulated for each u-cDSP inherently consider the compromise of the tradeoffs between the each objective function. With the goal of minimizing the

deviation of the expected utility from the ideal value, parameters that provide the best values for this overall objective are chosen while maintaining consistency with the designer's preferences. With this infusion, designers are now given the opportunity and ability to model multiple objectives each decision stage of the PPCTM. The infusion of the formulation of the u-cDSP is illustrated in an application to the cantilever beam tutorial problem in Box 3.9.

Box 3.9 – Defining a Multi-Attribute u-cDSP for each Decision Stage of the Cantilever Beam Tutorial Problem

There exist three modes of managing product customization for the cantilever beam example: customization of beam length, standardization of the cross-section, and standardization of the material (Box 2.3). After consideration the engineers at CustomTech decided that the most effective hierarchic arrangement is to first use customization of beam length (ΔL_1), then standardization of the cross-section (ΔP_1), and finally standardization of the material (ΔP_2) (Box 2.4). With the order of decisions established, the u-cDSP for each stage is formulated.

The formulation of each u-cDSP follows the four steps outlined in Figure 3.10. The designers at CustomTech quantitatively and qualitatively define their preferences (Box 2.8) and develop an expected utility function (Box 2.9). The system goals are then formulated (Box 2.10), followed by the formulation of the deviation function (Box 2.11). The resulting u-cDSP for each stage is presented below:

<u>Decision 1: ΔL_1</u>

Find: The value of decision variable ΔL_1
The deviation variables, $d_{i,1}^-$ and $d_{i,1}^+$

Satisfy: Bounds: $\Delta L_1 \leq L_{\max}$

Constraints: $\sigma_u \geq \dfrac{PL}{S}$

$$d_{i,1}^{-}, d_{i,1}^{+} \geq 0$$

$$d_{i,1}^{-} \square d_{i,1}^{+} = 0$$

Goals: $E[u(C_1)] + d_{C,1}^{-} - d_{C,1}^{+} = 1$

$$E[u(w_1)] + d_{w,1}^{-} - d_{w,1}^{+} = 1$$

Minimize: $Z_1 = 1 - U_1 = k_C(d_{C,1}^{-} + d_{C,1}^{+}) + k_w(d_{w,1}^{-} + d_{w,1}^{+})$; where $U_1 = k_C u(C_1) + k_w u(w_1)$

<u>Decision 2: ΔP_1</u>

Given: The value of ΔL_1

Find: The value of decision variable ΔP_1

The deviation variables, $d_{i,2}^{-}$ and $d_{i,2}^{+}$

Satisfy: Bounds: $\Delta P_1 \leq P_{\max}$

Constraints: $\sigma_u \geq \dfrac{PL}{S}$

$$d_{i,2}^{-}, d_{i,2}^{+} \geq 0$$

$$d_{i,2}^{-} \square d_{i,2}^{+} = 0$$

Goals: $E[u(C_2)] + d_{C,2}^{-} - d_{C,2}^{+} = 1$

$$E[u(w_2)] + d_{w,2}^{-} - d_{w,2}^{+} = 1$$

Minimize: $Z_2 = 1 - U_2 = k_C(d_{C,2}^{-} + d_{C,2}^{+}) + k_w(d_{w,2}^{-} + d_{w,2}^{+})$;where $U_2 = k_C u(C_2) + k_w u(w_2)$

<u>Decision 3: ΔP_2</u>

Given: The value of ΔL_1, ΔP_1

Find: The value of decision variable ΔP_2

The deviation variables, $d_{i,3}^{-}$ and $d_{i,3}^{+}$

Satisfy: Bounds: $\Delta P_1 \leq \Delta P_2 \leq P_{\max}$

Constraints: $\sigma_u \geq \dfrac{PL}{S}$

$$d_{i,3}^{-}, d_{i,3}^{+} \geq 0$$

$$d_{i,3}^{-} \square d_{i,3}^{+} = 0$$

Goals: $E[u(C_3)] + d_{C,3}^{-} - d_{C,3}^{+} = 1$

$$E[u(w_3)] + d_{w,3}^{-} - d_{w,3}^{+} = 1$$

Minimize: $Z_3 = 1 - U_3 = k_C(d_{C,3}^{-} + d_{C,3}^{+}) + k_w(d_{w,3}^{-} + d_{w,3}^{+})$;where $U_3 = k_C u(C_3) + k_w u(w_3)$

3.3.2 A Critical Analysis of the Infusion of the u-cDSP into the PPCTM

To increase the complexity of the types of problems that can be solved with the PPCTM, it must be augmented in order allow designers to make decisions in the presence of multiple objectives (Section 2.1.3). The infusion of the utility-based compromise DSP decision construct into the PPCTM provides a designer with the ability and opportunity to handle multiple objectives.

The u-cDSP is a decision construct that can quantitatively capture designer preferences, model non-linear preferences, and also handle risk & uncertainty in system goals, variables, and bounds. It is noted, however, that the inclusion of risk and uncertainty analysis is beyond the scope of this work and is suggested as future work (see Chapter 6). As a result, the author recognizes that the use of utility theory in this particular work might be too complex; the tenets of value theory (Keeney et al., 1976) alone would be sufficient for the solution. It is the author's intent, however, to include the framework for utility theory so that it's inclusion in the future is easier.

The focus in this chapter is to present the augmented PPCTM in a manner in which theoretical structural validity is developed. Theoretical structural validity involves accepting the individual constructs and premises constituting the method as well as the internal consistency of their assembly into a method. From the above discussion, it can be argued that following the assumption that if both the c-uDSP and the PPCTM are internally consistent, than the infusion of one into the other is internally consistent as a whole. This assumption cannot be entirely made, as one must also verify that the c-uDSP is being correctly infused into the PPCTM without violating any of its tenets. This point can be verified, however, as seen in the above description, the u-cDSP's multi-stage

decision construct can be fit into the space provided by the removal of the previous dynamic programming references of the PPCTM.

While the limitations of the PPCTM have been repeated throughout this thesis, it should be noted that it was developed in an open manner so as to invite future augmentation. Hernandez recognized the need for the ability to model non-uniform demand and the need for multi-objective decision-making; as such, the PPCTM was built almost modularly in order to openly accept these new tools. The augmented method, the focus in this thesis, is presented in the next section.

3.4 A METHOD FOR DESIGNING PRODUCT AND PROCESS PLATFORMS IN A GEOMETRIC SPACE OF NON-UNIFORM DEMAND

There are three research thrusts in this thesis (Figure 1.6): the augmentation of the PPCTM to handle non-uniform demand, multiple objectives, and the design of the manufacturing process of a customized product. The augmentations presented in Chapters 2 and 3 – non-uniform demand modeling techniques, the utility-based compromise Decision Support Problem, and production process platform concepts – provide the constructs necessary to alleviate the limitations of the PPCTM (see Figure 3.11, a response to Figure 1.6).

The result of the augmentation is a six-step method (Figure 3.12) that enables a designer to methodically develop both product and process platforms for customizable products in a market space of non-uniform demand with the presence of multiple objectives.

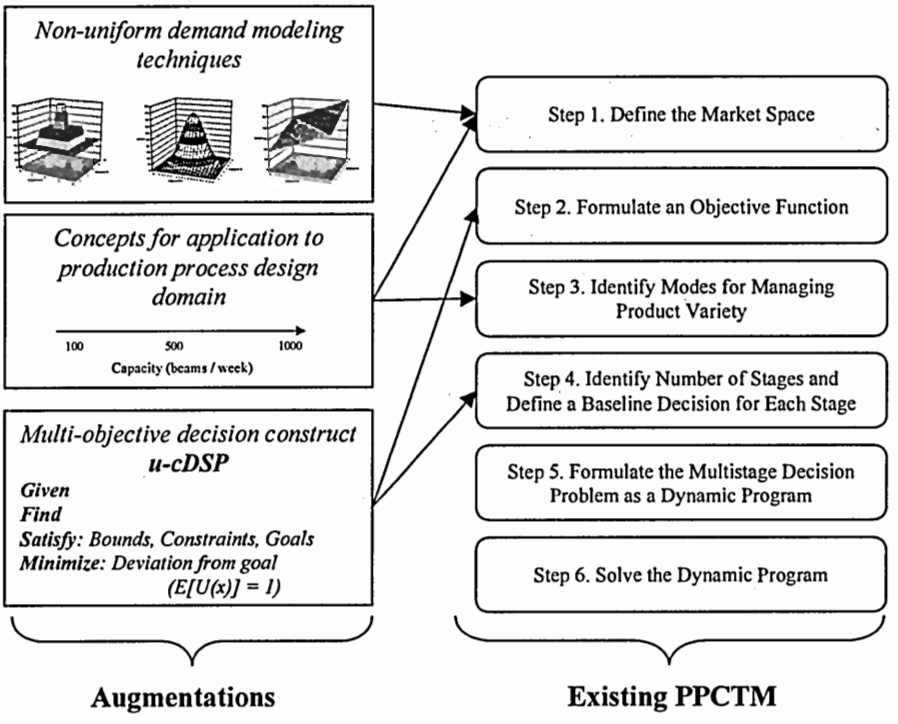


Figure 3.11 – Augmenting the Product Platform Constructal Theory Method

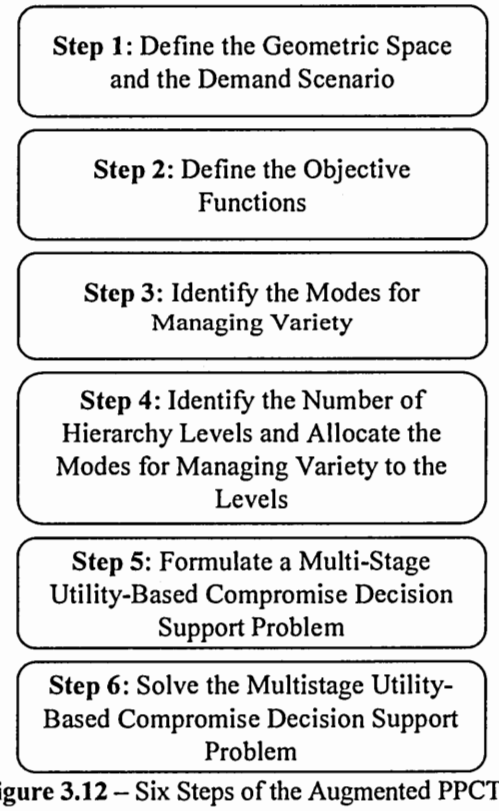


Figure 3.12 – Six Steps of the Augmented PPCTM

3.4.1 Step 1: Define the Geometric Space and Demand Scenario

The framework of the augmented PPCTM is general enough to enable a designer to develop both a product and a production process platform. Since this first step involves the establishment of the context of the problem, the exact terms used in its definition are dependent on which type of platform is being developed. As such, there is a separate description provided for the development of a product platform and for a process platform.

Product Platform Design: Defining the Customization Space

For the development of a product platform, this first step is almost identical to that of the PPCTM. A space for customization (previously defined as a market space, Section 2.1.2) is defined by three components:

- the identification of the product parameters that will be varied according to the customer demands,
- the range of variety that will be offered for each parameter, and
- the analysis of the demand of the market.

The number of dimensions of the geometric space is directly related to the number of product parameters in which variety will be offered. Given n independent product requirements $r_1, r_2, \ldots r_N$, a n-dimensional space for customization is defined, $M_n = \{(r_1, r_2, \ldots, r_n)\}$. The range of each varied parameter determines the bounds of each dimension of the geometric space.

Finally, a designer is required to define the current demand of the market. The distribution of the demand of each product variant is modeled using one of the techniques presented in Section 3.1 (discrete, linear continuous, or normal distribution).

A result of the implementation of one of these modeling techniques is a formulation of demand as a function of the varied product parameters.

It should be noted that the original term "market space" is changed to "space of customization" in this augmented PPCTM. This is done for two reasons:

- the term "space of customization" effectively captures the notion that this is a geometric space that represents variants of a customized product
- the term "market space" is used in other facets of research; this switch will keep these concepts separate and will also assist with the creation of a specific lexicon for this research direction.

The use of the term "space of customization" has been used in the latest publications of Hernandez's continuing work on the PPCTM (Hernandez et al., 2002; Hernandez et al., 2003).

Process Platform Design: Defining the Space of Capacity

The geometric space to be defined in the formulation of the augmented PPCTM for the development of a process platform is the space of capacity (Section 3.2.2). The definition of an appropriate space of capacity involves the identification of the range of production capacity that the manufacturing enterprise wishes to offer. The resulting space of capacity will be a one-dimensional space bounded by the range of capacity. This space represents the range of capacity that the process platform will satisfy; it also inherently captures the problem of non-uniform demand found in the design of a production process.

3.4.2 Step 2: Define the Objective Functions

Through the infusion of the utility-based compromise Decision Support Problem into the PPCTM, a designer has the opportunity to define multiple objectives (Section 3.3.1).

The objective functions represent parameters to be improved by the use of the method. Examples of appropriate objective function for product design include the minimization of cost, maximization of profit, or the improvement of any design parameter such as minimization of mass, maximization of thermal efficiency, or the maximization of strength. For design of a production process, appropriate examples include minimization of cost, maximization of throughput, or maximization of quality.

An objective function can be formulated in one of two ways: through a discretized analysis of the space using a summation equation (Equation 3.19), or through a continuous analysis of the space using an integral (Equation 3.20).

$$o = \sum_{i=r_{1,\min}}^{r_{1,\max}} \sum_{j=r_{2,\min}}^{r_{2,\max}} \cdots \sum_{k=r_{N,\min}}^{r_{N,\max}} O(r_{1,i}, r_{2,j}, \ldots r_{N,k}) \qquad [3.11]$$

$$o = \int_{r_{N,\min}}^{r_{N,\max}} \cdots \int_{r_{2,\min}}^{r_{2,\max}} \int_{r_{1,\min}}^{r_{1,\max}} O(r_1, r_2, \ldots r_N) dr_1 dr_2 \ldots dr_N \qquad [3.12]$$

where the *min* and *max* subscripts refer to the upper and lower bounds, respectively, of each dimension of the space. It should be noted that the space of customization of each of the example problems in this thesis are analyzed through a discretized analysis.

If the objective to be minimized or maximized is to be formulated as an average across the entire product family, the objective function is divided by the total demand of the subspace (shown in Section 3.1). This total demand is determined by summing the demand function developed in the previous stage across the space.

3.4.3 Step 3: Identify the Modes for Managing Variety

Similar to Step 1 presented in earlier (Section 3.4.1), the lexicon involved with the formulation of this step is different for product and process platform design.

Product Platform Design: The Modes for Managing Product Variety

This step is identical to that of the original formulation of the PPCTM (Section 2.1.2). This step involves the consideration of how a product will be varied in order to satisfy the customized specifications of the market space. Modes of managing product customization are the linking mechanism between the product variants that compose the product family. Examples of modes for managing product customization include component commonalization, dimensional comonalization, modularity, and standardization. There is no rigid structure to the formulation of these modes; this step instead involves brainstorming and decision-making between design and manufacturing.

Process Platform Design: The Modes for Managing Process Variety

For this step, a designer identifies appropriate methods for providing variety in capacity for the manufacturing system. Modes of managing process customization are the linking mechanism between the different production capacity requirements that compose a process family (Section 3.2.3). Examples of modes for managing process customization include, but are not limited to, standardization of specific process parameters, commonalization of the batch size, standardization of machine type, and modular combinations of machines (i.e., increasing the number of machines in the production system).

3.4.4 Step 4: Identify the Number of Hierarchy Levels and Allocate the Modes of Managing Variety to the Levels

In the fourth step, it is established how and when each mode of managing customization is used. If there are N modes for managing product variety, up to N hierarchic levels, one per mode, can be created. If there are fewer hierarchic levels chosen than there are modes, then more than one mode must be used in at least one of the hierarchic levels. The ordering of the modes typically follows the pattern in which modes that are capable of achieving the smallest variations in the varied design parameters are used at the lower levels of the hierarchy (i.e., before modes that can only achieve large variations in the design parameters to be varied) (Section 2.1.2).

This step is formulated differently than that of the original PPCTM. This is done so more importance is placed on this portion of the method. Scenarios arise when the selection of the proper arrangement of the hierarchy is not immediately evident. The separation of this decision from the formulation of the decision construct also places the appropriate emphasis on the infusion of the u-cDSP.

3.4.5 Step 5: Formulate a Multi-Stage Utility-Based Compromise Decision Support Problem

Following the tenets of constructal theory, the determination of the range of the modes of managing variety that compose a level of the hierarchy represents one stage in a multi-stage decision. With the order of the use of the modes established, a designer defines a utility-based compromise Decision Support Problem for each decision stage in this step.

The baseline decision of each stage is the determination of the size and shape of each subspace. This is represented by the determination of the range of commonality of

each mode for managing variety, $\Delta r(i)$. For a problem with N varied parameters, the decision variables for any stage i, are

$$\Delta r(i) = [\Delta r_1(i), \Delta r_2(i), \ldots, \Delta r_N(i)] \qquad [3.13]$$

In the formulation of each u-cDSP, there exists a constraint on the selection of the range of commonality for each mode of managing so that a hierarchic design is achieved:

$$\Delta r_j(i+1) \geq \Delta r_j(i) \qquad [3.14]$$

Each u-cDSP is formulated to minimize the deviation of the expected utility from the ideal value. The expected utility function is a multi-attribute utility function that is the sum of weighted individual functions of preference for tradeoffs of multiple objectives (Section 2.2.3). In this thesis, all designers' preferences are modeled as risk averse and the expected utility function is formulated as an additive multi-attribute utility function.

The formulation of each utility-based compromise Decision Support Problem follows the four steps presented in Section 2.2.3 (its infusion into this fifth step of the PPCTM is presented graphically in Figure 3.10). First a utility function for each of the objectives is formulated by qualitatively and quantitatively assessing the preferences of the designers. These individual utility functions are then combined into a multi-attribute utility function as a weighted average of the individual utilities. Finally, goal and deviation functions are developed for each stage. The infusion of these steps to the PPCTM is described in Section 3.3.1 (specifically Box 3.9). A typical u-cDSP formulation is presented in Figure 3.13.

For Each Stage i

Given: The N-dimensional market space $M^N = \{(r_1, r_2, \ldots r_N)\}$
The decision variables of the previous stages $\Delta r(1), \ldots, \Delta r(i\text{-}1)$
The modes of managing product variety to be utilized at Stage i

Find: The value of decision variable $x(i)=[\Delta r_1(i), \Delta r_2(i), \Delta r_N(i)]$
The deviation variables, $d^-_{x,i}$ and $d^+_{x,i}$

Satisfy: Bounds: $\Delta r_{j,\min}(i) \le \Delta r_j(i) \le \Delta r_{j,\max}$

Constraints: $\Delta r_j(i) \ge \Delta r_j(i-1)$

$d^-_{x,i}, d^+_{x,i} \ge 0$

$d^-_{x,i} \square d^+_{x,i} = 0$

Goals: $E[u(o_{x,i})] + d^-_{x,i} + d^+_{x,i} = 1$

Minimize: $Z_i = 1 - U_i = \sum_{x=1}^{y} k_x (d^-_{x,i} + d^+_{x,i})$; where $U_i = \sum_{x=i}^{y} k_{x,i} u(o_{x,i})$

Figure 3.13 – Formulation of the Multi-Stage Utility-Based Compromise Decision Support Problem

3.4.6 Step 6: Solve the Multi-Stage Utility-Based Compromise Decision Support Problem

The final step of the augmented PPCTM is the solution of the multi-stage utility-based compromise Decision Support Problem formulated in the previous step. Fundamentally, the key decision of the application of the PPCTM is the determination of the values of the ranges of each mode for managing product variety ($\Delta r(i)$) that provide the smallest deviation of the expected utility function from the ideal value of 1 (Figure 3.13).

The solution method proposed in the formulation of the original PPCTM is the use of dynamic programming to approximate the relationship between each decision stage (Section 2.1.2). The implementation of dynamic programming in the context of the PPCTM requires a designer to first "move backwards" through the decision stages in

order to develop response surfaces that approximate different ranges of each mode to certain objectives. The problem is then solved by "moving forwards" through the each response surface to calculate the objective function (Figure 2.18).

This solution technique becomes extremely tedious with the presence of multiple objectives (i.e., more than one), multiple modes of managing variety (i.e., more than three), and multiple hierarchy levels (i.e., more than three). Although the use of response surfaces provides adequate approximations, the development of each response surface requires an exhaustive search of combinations of design parameters and their influence on the objectives. Following the completion of his dissertation, Hernandez has moved away from the emphasis on dynamic programming, and has moved to a more generic description of this sixth step, "Solve the Multi-Stage Optimization Problem" (Hernandez et al., 2002; Hernandez et al., 2003). In these recent publications, Hernandez has turned to exhaustive searches to solve the multi-stage problem (although for admittedly simpler examples).

Following this trend, this sixth step involves the implementation of "an appropriate solution algorithm" for the determination of the main decision variables of the multi-stage u-cDSP. The framework in which the solution algorithm is implemented is presented in Figure 3.14.

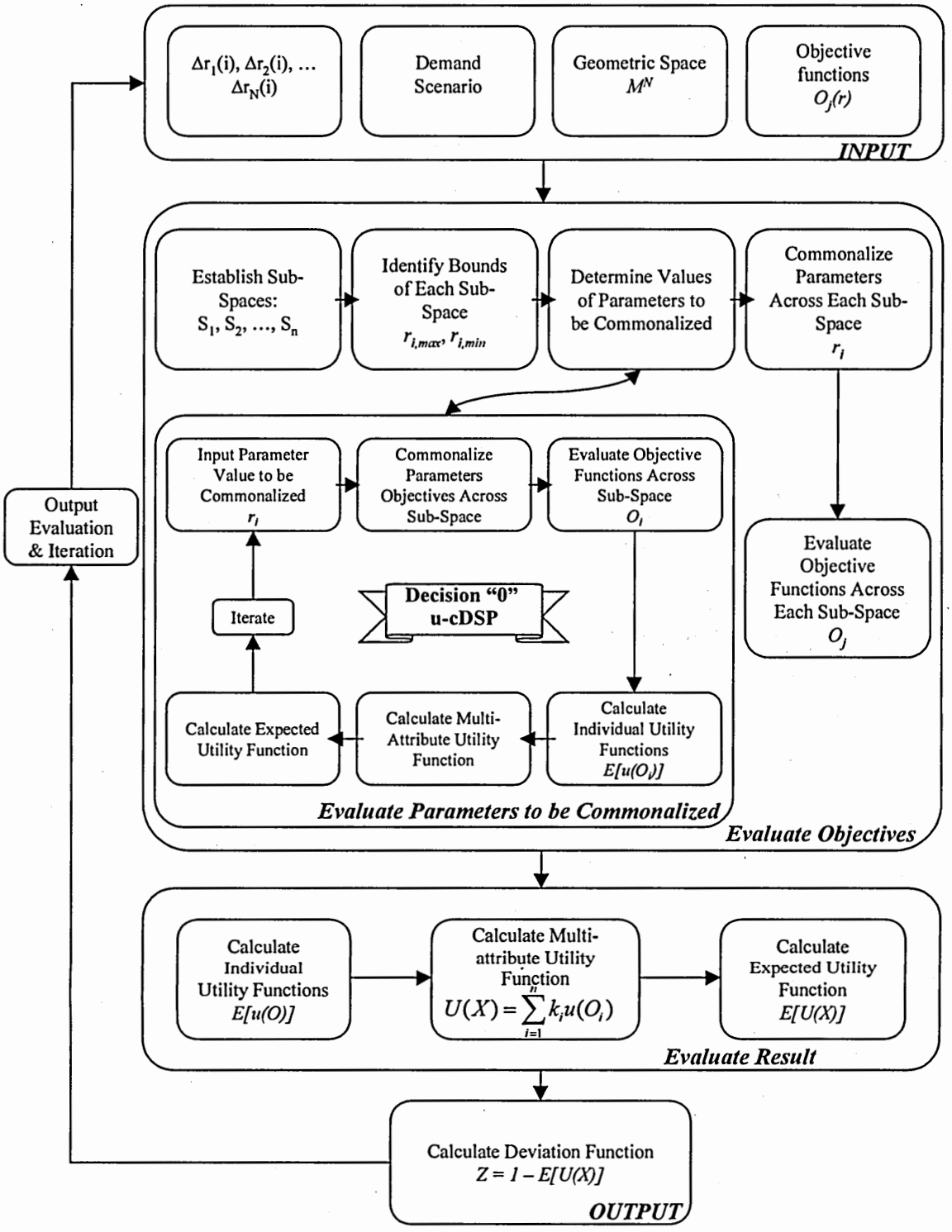


Figure 3.14 – General Solution Method for the Augmented Product Platform Constructal Theory Method

The general concept of *searching* for a solution (i.e., the decision variables $\Delta r(i)$) via the process of inputting a guess, evaluating the deviation function, and iterating is illustrated in Figure 3.14. Specifically, the following general steps are followed:

- The demand scenario, the geometric space, the formulation of the objective functions, and the implementation of the modes for managing variety are known (from the formulation of the u-cDSP of the previous step).
- A guess of the best combination of ranges for each mode for managing variety is made.
- From these ranges, each sub-space is defined.
- From the size and shape of each sub-space, the bounds of each space are determined.
- The values of the process parameters to be commonalized for each sub-space are determined. This involves an internal decision, named "Decision 0"
    - For most applications of the PPCTM, especially for the development of product platforms, the determination of the values of parameters to be commonalized does not require the formulation of a u-cDSP. This is because, typically, design constraints dictate which value will provide the highest utility for the sub-space.
    - For more problems with multiple conflicting objectives, or for problems in which design constraints do not provide an obvious value to be commonalized, a separate u-cDSP must be formulated in order to make a decision. This u-cDSP is extremely similar to that which is formulated for the

entire problem - the only difference is that this decision only applies to a single sub-space. The goal of this u-cDSP is the determination of the values of the process parameters that should be commonalized across the sub-space which provide the largest expected utility for that space. The specific manner in which this u-cDSP is solved is to be determined by a designer.

- Within the bounds of each space, the appropriate parameters are commonalized, and the objective functions across each sub-space are calculated.
- Moving into the "Evaluate Result" stage, the expected utility of the entire geometric space is evaluated by calculating the individual utility functions and formulating a multi-attribute utility function.
- The "output" of this algorithm is the calculation of the deviation function for a specific combination of values of the decision variables.
- This output is then compared to outputs of other inputs. This process is iterated until a designer is confident that the best solution has been found (the stopping criterion is dependent on the solution algorithm chosen).

This is a general technique in which the decision variables for the multi-stage u-cDSP formulated in Step 5 can be determined. This technique can be implemented with any appropriate solution technique (i.e., exhaustive search, Generalized Reduced Gradient, Sequential Quadratic Programming, etc.).

This solution method is not presented as a significant contribution towards the implementation of the PPCTM. The formulation of "the most efficient" solution method is not presented as a research goal in this thesis (Section 1.3). While this solution method is not the most eloquent, it is rigorous and is capable of providing empirical structural

validity to the research goals of this work: the augmentation of the PPCTM to provide its user the ability to cope with non-uniform demand, multiple objectives, and the design of a production process for a customized product.

The result of this augmented process is the hierarchic organization of multiple modes of managing customization for the development of a product or a process family in a market of non-uniform demand. Specifically, the formulation of the multiple u-cDSPs, provides a designer the ability to select the range of commonality for each mode of managing customization in order to provide the best compromise for multiple conflicting objectives.

3.5 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD

The focus in this thesis is the augmentation of the PPCTM to facilitate three objectives:

- to enable the user of the PPCTM to model non-uniform demand
- to abstract the theory of the PPCTM to the realm of process design, namely in the development of process platforms
- to infuse a decision construct into the PPCTM so that a designer can cope with multiple objectives

In Chapter 2, an explanation of the PPCTM and its underlying constructs is presented along with the description of the utility-based compromise Decision Support Problem. The role of this third chapter is to:

i. present potential methods for modeling non-uniform demand,

ii. discuss how the PPCTM is applicable to the realm of process design,

iii. detail the infusion of the u-cDSP into the PPCTM, and

iv. present the formulation of the augmented PPCTM.

For role (i), three different methods of modeling demand that are used in this thesis is presented in Section 3.1. It is understood that these are not all of the different possible ways in which to model demand; instead these are the most applicable for this body of work. It should be noted that, conceptually, the resulting augmented PPCTM has the ability to handle all methods of demand modeling.

For role (ii), the case for the need of the development of a production process platform is made in Section 3.2. Three key definitions are provided in order to abstract the PPCTM to the realm of process design (process platform, space of capacity, and modes for managing process variety). Finally, a brief overview of how the PPCTM maps to the development of a process platform is presented.

For role (iii), the detail of how the u-cDSP can be infused into the PPCTM is presented in Section 3.3. With the understanding of its underlying constructs (from Chapter 2), the multi-stage decision problem of the PPCTM is transformed into the formulation of a utility-based compromise Decision Support Problem. The result is a re-formulation of the PPCTM so that a designer may be able to use it to make decisions in lieu of the presence of multiple objectives.

Finally, in Section 3.4, the augmented PPCTM is presented. The resulting six-step method enables a designer to develop either a product or a process platform for customizable products in the face of non-uniform demand and multiple objectives.

The larger role of this chapter is the strengthening of theoretical structural validity (Figure 3.15).

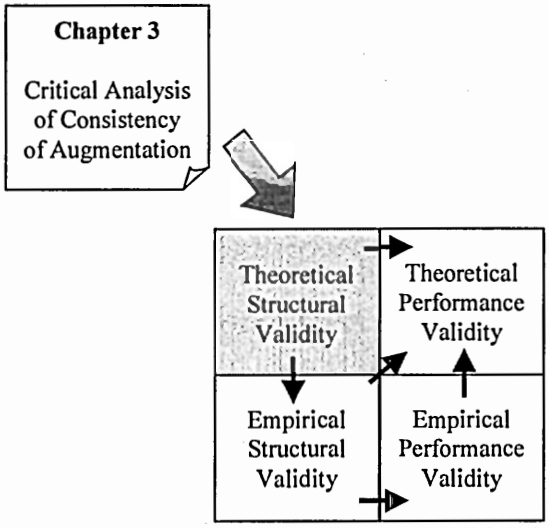


Figure 3.15 – Theoretical Structural Validation Established in Chapter 3

The case for theoretical structural validity began in Chapter 2 with the presentation of the underlying theoretical constructs of the components of the resulting augmented PPCTM. In this chapter, this portion of the validity square is further strengthened though the presentation of how these separate constructs are put together. Returning to the four roles of this chapter, the foundation for theoretical structural validity is observed:

(i) The three methods of modeling demand are valid, as they are simple mathematical functions. The application of these functions to the modeling of demand has been recorded as appropriate (Section 3.1). The infusion of these modeling methods into the PPCTM is deemed appropriate as it provides only a slight change in the formulation of the method.

(ii) The application of the PPCTM into the realm of process design is also structurally sound. It is shown how the development of a process can be abstracted as a problem of optimal access in a similar fashion that Hernandez did in his abstraction of the product design process (Section 3.2).

(iii) The infusion of the u-cDSP into the PPCTM requires only a simple substitution. The fundamental axioms supporting the u-cDSP and the PPCTM

remain intact through this infusion, thus it is theoretically sound (Section 3.3.2).

With the reasoning behind the design of product and process platforms for customizable products established (Chapter 1), the details of the core constructs of the PPCTM and the u-cDSP discussed (Chapter 2), and the formulation of the augmented PPCTM presented, the focus in this thesis has almost been achieved. The application of this augmented method, and the resulting claims for empirical structural and empirical performance, are presented in Chapters 4 and 5 in order to complete the work. In Chapter 4 the augmented PPCTM will be applied to the design of a product platform for a line of customizable pressure vessels. In Chapter 5, the effectiveness of the PPCTM as a process platform design method is tested through the development of a process platform for a line of customizable hearing aid shells.

CHAPTER IV

CUSTOMIZED PRODUCT PLATFORM DESIGN: PRESSURE VESSELS

A method for designing a family of products and processes for customization in a market with non-uniform demand is presented in Chapter 2. The application of this method in the realm of *product* platform design for customized products is the focus in this chapter. Specifically, through the application of the augmented PPCTM, multiple modes of managing product variety are used to design a family of customizable pressure vessels for a series of non-uniform demand scenarios.

The pressure vessel example problem is presented in Hernandez's previous applications of the PPCTM (Hernandez, 2001; Hernandez et al., 2003). As such, it is an appropriate example for the validation of the augmentations of his work. The pressure vessel example is a classic mechanical engineering problem. It is familiar to most engineers and has only a few design equations. As such, it is a perfect "first example" as it allows the reader to focus on the method without having to worry about the engineering design details of the problem at hand. The fundamental problem description and development presented in this work is similar to that done by Hernandez, and is cited where appropriate.

In this chapter the empirical structural and performance validity is established, as shown in Figure 4.1. By applying the augmented PPCTM (Chapter 3) to this example problem, confidence in the appropriateness of the example and confidence in the usefulness of the method will be established.

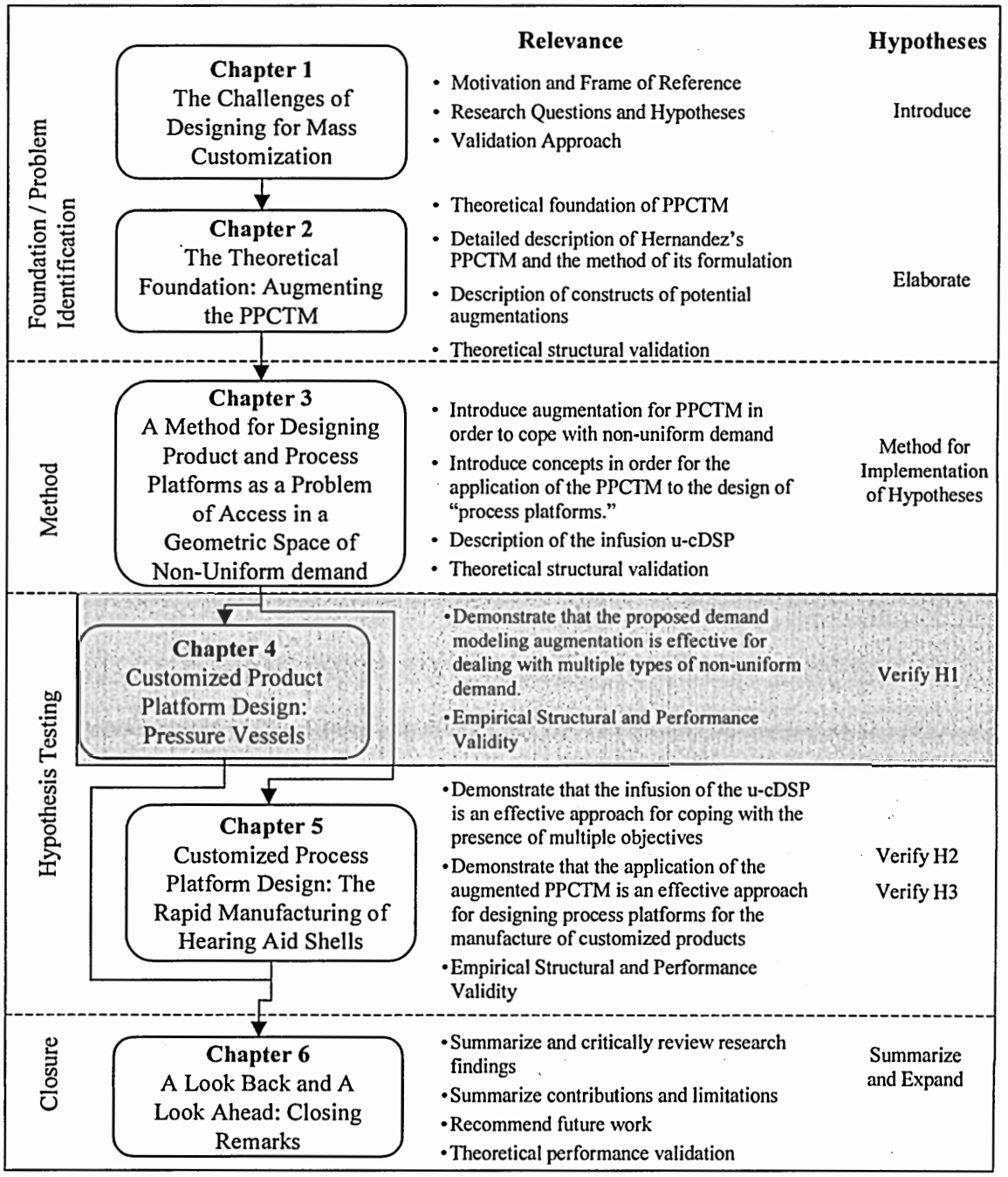


Figure 4.1 – The Role of Chapter 4 in this Thesis

Validation aside, the core focus in this chapter is to provide empirical support for Hypothesis 1 (from Section 1.3):

> **Q1**: How can the Product Platform Constructal Theory Method be augmented in order to cope with the development of product platforms in a market of non-uniform demand?
> **H1**: *Modeling the demand as a property of the geometric market space, a designer can successfully solve the non-uniform optimal access problem through the discretization of the space.*

The pressure vessel example problem is introduced in Section 4.1. The augmented PPCTM is applied to the example in Section 4.2. The description of the solution method and the presentation of results are then presented in Section 4.3. Finally, in Section 4.4, a summary of the chapter and a preview of the next chapter are given.

4.1 THE PROBLEM OF OFFERING CUSTOMIZED PRESSURE VESSELS IN A MARKET OF NON-UNIFORM DEMAND

Consider a manufacturer of pressure vessels. This manufacturer seeks a competitive advantage over other leading manufacturing enterprises through offering customized vessels. By providing customers the opportunity to choose a pressure vessel suited to their specific needs (a choice of any volume and pressure within the set market bounds), it is assumed that the resulting customized product family (and all of its variants) will be of high demand and therefore profitable.

There are three key assumptions in the formulation of the customizable pressure vessel problem:

1. *A fixed market space*: the use of the PPCTM to design a product platform is suited only for the market space presented. It is assumed that the market space will not contract or expand over time. This assumption does not apply only to

the bounds of the market space; it also applies to the demand scenario initially presented.

2. *Scalable product components*: The manufacturer seeks to develop a product family of customizable vessels by scaling product components. Offering variety through altering product functionality is not considered in this example problem.
3. *Complete and certain information*: All relevant information and models needed to apply the augmented PPCTM are available, complete, and certain. The roles of uncertainty and risk are not taken into account in the formulation of the problem. Conceptually, the infusion of the u-cDSP into the PPCTM should provide a designer with the power to model uncertainty; however, this is beyond the scope of this work.

The detailed description of the pressure vessel market is presented in Section 4.1.1. The product model that will is used in the application of the method is then discussed in Section 4.1.2. The establishment of context provides the foundation on which the application of the augmented PPCTM is presented.

4.1.1 Pressure Vessel Market Description

The manufacturer wishes to offer a line of customizable pressure vessels that range in volume from 10 to 30 m^3, and in pressure from 10 to 30 MPa. It is assumed that all combinations of pressure and volume are technically feasible within the bounds given. It is also assumed that offering customers the opportunity to order a customized pressure vessel from these two specified ranges is a profitable and viable competitive advantage.

As explained above, the pressure vessel market is fixed - the market space is not expected to contract or expand in time. While the size and shape of the market space is uniform over time, the demand for product variants within the market is not. In the implementation of this example problem, the ability of the augmented PPCTM to handle any type of non-uniform demand is tested against ten different demand scenarios.

For this example, demand is modeled with three different techniques: discrete changes in demand, linear changes in demand, and a normal distribution of demand. From these three different techniques, ten different demand scenarios are created in order to test the method's ability to cope with non-uniform demand. These different demand scenarios are discussed in Sections 4.2.1 and 4.3.3.

4.1.2 Pressure Vessel Product Model

For this example, it is assumed that the conceptual design phase and the beginning of the embodiment phase of the pressure vessel have already been completed. Each vessel consists of a cylindrical container capped on both ends by hemispherical heads. A schematic of the basic design is presented below:

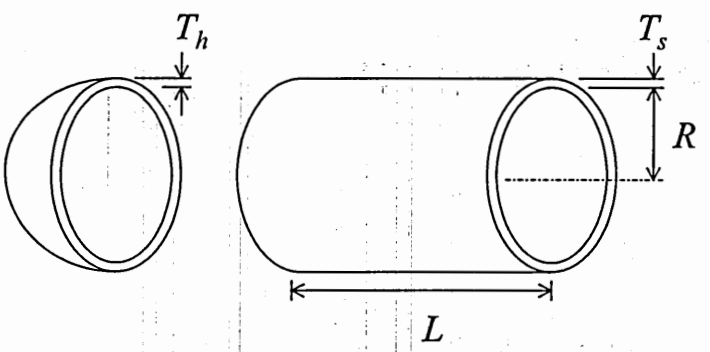


Figure 4.2 - Schematic of Pressure Vessel Embodiment (Hernandez, 2001)

The manufacturer wants to offer a range of pressure and volume combinations. Specifically, the manufacturer wants to specify for each possible combination of pressure and volume the following design variables: length (L), radius (R), and the head and shell

thickness (T_h and T_s). The manufacturer has determined that there is only one objective in the design of the product platform: the maximization of the average profit (the average profit is defined in Section 4.2.2).

The pressure vessels are produced from carbon steel ASME SA 203 grade B. Sheets of this material are available in thicknesses ranging between 6.35mm and 76.2mm. Available equipment limits the maximum radius to 1.5m and the maximum length to 7m.

The design of the vessels must satisfy the following constraints on the minimum allowable thickness of the shell and for a given pressure (Bednar, 1986):

$$T_s \geq \left(\frac{P}{\sigma_y - 0.6P} \right) R \qquad [4.1]$$

$$T_h \geq \left(\frac{P}{2\sigma_y - 0.2P} \right) R \qquad [4.2]$$

where P is the pressure, σ_y is the yield strength of the material (1077 MPa).

The objective in developing this product platform is to maximize the average profit of the entire market. As a result, a means of modeling the selling price and the cost of the vessels is presented. The analysis of the cost is similar to that used by Hernandez used in his original application of the PPCTM to the pressure vessel example (Hernandez, 2001).

The total cost of the manufacturing of the pressure vessel (ignoring all labor and plant costs) includes four components: material cost, welding cost, order cost, and equipment cost.

The material cost is determined by the amount of material purchased to build the vessel. This cost includes both the cost of the material used in each vessel and the cost of

the material wasted by cutting the raw steel plates to the required dimensions. The cost of material ($C_{material}$) is therefore given by:

$$C_{material} = 2\pi\rho(C_s R T_s L + C_h R^2 T_h) + C_{waste} \qquad [4.3]$$

where ρ is the density of the material (7800 kg/m^3), C_s is the cost per kg of processed shell steel (\$0.8 per kg), and C_h is the cost per kg of forged steel for the head (\$2 per kg). The cost of wasted material (C_{waste}) is given by:

$$C_{waste} = 2\pi\rho C_p T_s R(L_o - L) \qquad [4.4]$$

where L_o and C_p are the length and cost per kg (\$0.3 per kg), respectively of a raw steel plate.

All welds in the manufacturing process are single-welded butt joints with a backing strip. The welding cost (C_{weld}) is composed of the cost of the longitudinal weld and the cost of the circumferential weld. The longitudinal welding cost ($C_{longweld}$) is given as:

$$C_{longweld} = V_l \rho C_w \qquad [4.5]$$

where the volume of the welding material, V_l is given by

$$V_l = 2\pi\left(\frac{T_s}{\cos 30^\circ}\right)^2\left(\frac{60}{360}\right)L = \frac{4}{9}\pi T_s^2 L \qquad [4.6]$$

and C_w is the approximate cost of welding material (15 \$/kg hand welded).

The circumferential welding cost ($C_{circweld}$)is given as:

$$C_{circweld} = V_s \rho C_w \qquad [4.7]$$

where the volume of the welding material, V_s, is given by

$$V_s = 4\pi^2\left(\frac{T_s}{\cos 30^\circ}\right)^2\left(\frac{60}{360}\right)R = \frac{8}{9}\pi^2 T_s^2 R \qquad [4.8]$$

The total welding cost becomes

$$C_{weld} = 2\pi\rho\left(\frac{2}{9}C_w T_s^2 L + \frac{4}{9}C_w \pi T_s^2 R\right) \qquad [4.9]$$

The total cost for a single pressure vessel, without including equipment costs is therefore,

$$C(L,R,T_s,T_h) = 2\pi\rho\left(C_s R T_s L + C_h R^2 T_h + \frac{2}{9}C_w T_s^2 L + \frac{4}{9}C_w \pi T_s^2 R + C_p T_s R\left(L_o - L\right)\right) \qquad [4.10]$$

There is also a cost associated with ordering the material, C_{order}. Each time an order for raw material is placed, a fee of \$250 is assessed in order to cover shipping, handling, and stocking in the inventory. This cost is based on the number of different sized sheets of raw material is order; it is not related to the quantity of sheets ordered. This cost is tabulated using Equation 4.11.

$$C_{order} = \sum_{i=1}^{m} \$250 \qquad [4.11]$$

where m is the number of distinct sheets of metal required for the manufacturing process (i.e, the number of distinct values of L_o).

The cost of purchasing manufacturing equipment (C_{equip}), namely the forging presses and the associated dies to make the heads of the vessels, is evaluated with

$$C_{equip} = \sum_{p=1}^{N_p}\left(500000 + 50000R_p\right) \qquad [4.12]$$

where N_p is the number of presses needed, and R_p is the radius of the die used with each press. The cost equation presented in Equation 4.10 is the total cost of producing one vessel; the ordering cost and equipment cost (Equations 4.11 and 4.12, respectively) are calculated for the entire family of vessels and their required manufacturing needs. It is

assumed that each forging press has a very large production capacity; one press for each radius is sufficient to meet any demand.

With the cost model complete, a manner for determining the selling price of each product variant is determined in order to evaluate the total profit. For this example the selling price is established as a function of volume and pressure:

$$Selling\ Price = C_v V + C_p P \qquad [4.13]$$

where C_v and C_p are the cost constants for volume (185 \$/m^3) and pressure (185 \$/MPa) respectively. Finally, total profit is established by summing the difference of the selling price and cost for all of the product variants, n, as well as subtracting the cost of the orders, and the cost of the manufacturing equipment needed for the production process.

$$Profit_{total} = \sum_{i}^{n} \left[SellingPrice_i - C_i(L, R, T_s, T_h) \right] - C_{order} - C_{equip} \qquad [4.14]$$

The six-step augmented PPCTM (Section 3.4) is applied to this example problem in Section 4.2. The focus in this example is the application of the augmented PPCTM to facilitate the development of a product platform for customizable pressure vessels in a market of non-uniform demand. The resultant product platform will consist of a set of baseline of design variable values (including raw material needs; i.e., radius of dies, and the length and thickness of steel plates) from which customized pressure vessels will be produced.

4.2 EXPRESSING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE

In this section the manner in which the augmented Product Platform Constructal Theory Method (PPCTM) is applied to the product platform design of a line of

customizable pressure vessels is presented. The six steps of the augmented PPCTM are presented in Figure 4.3.

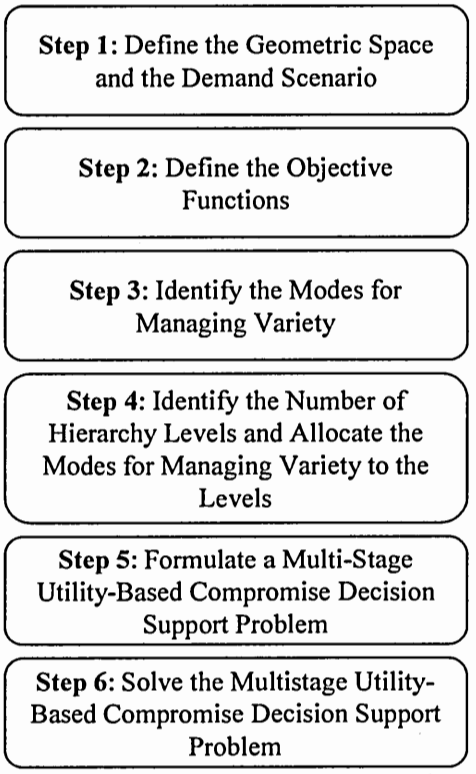


Figure 4.3 – The Six Steps of the Augmented PPCTM

In this section, the application of the first five steps is presented. The sixth step, the solution of the multistage utility-based compromise Decision Support Problem, is presented in Section 4.3 along with a discussion of the solution process and an analysis of the results.

4.2.1 Step 1: Define the Geometric Space and the Demand Scenario

As described in Section 4.1.1, there are two independent design specifications that characterize the desired product customization - the volume and the pressure of the pressure vessel. The volume range to be considered is 10 to 30 m^3, and the pressure range is 10 to 30 MPa. This results in a two-dimensional continuous space of customization as illustrated in Figure 4.4.

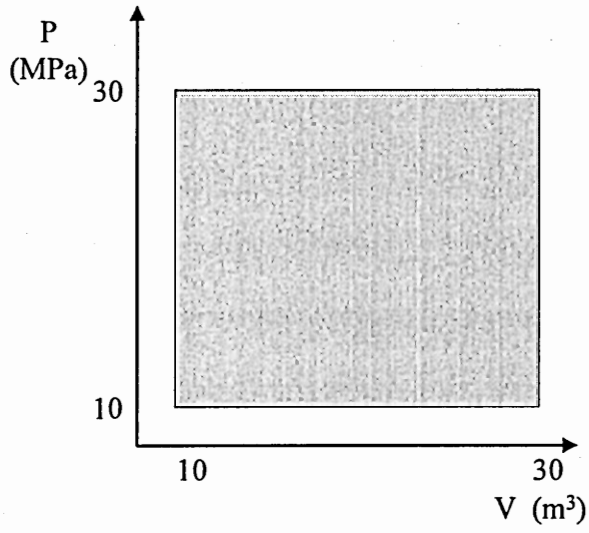


Figure 4.4 - Pressure Vessel Space of Customization

This space of customization is the representative geometric space for which the access to all points is to be optimized via constructal theory (Section 2.1.1).

If a designer were following the formulation of Hernandez's original PPCTM, this step would be complete. Since the formulation of this problem does not include Hernandez's assumption of uniform demand, a designer must choose an appropriate method for modeling demand before moving to the next step. For this example, three methods for modeling demand are tested (Section 3.1):

(1) Discrete Function, as presented in Section 3.1.1

(2) Linear Continuous Function, as presented in Section 3.1.2

(3) Two-Dimensional Gaussian Distribution, as presented in Section 3.1.3

The specific magnitude and arrangement of the values of demand are changed in each of these methods, totaling ten different demand scenarios. These results, and the lessons of product platform design abstracted from them, are presented in Section 4.3.

4.2.2 Step 2: Define the Objective Functions

As stated in the problem description in Section 4.1.2, the manufacturer of the line of customizable vessels wants to maximize the average profit of the products in the entire market space. The potential profit obtained from each product variant is estimated as the difference between the selling price (Equation 4.13) and the cost of the vessel (Equations 4.10 – 4.12. Following Equation 4.14, the average total profit is simply the summation of the product of the profit and the demand of the specific variants of the entire market space divided by the total demand of the market space. The formulation of the objective function for the solution of the method through the discretization of the market space is shown in Equation 4.15.

$$AvgProfit = \frac{1}{D_{tot}}\left[\left(\sum_{i=1}^{n} D_i(SellingPrice_i - C_i)\right) - C_{order} - C_{equip}\right] \qquad [4.15]$$

where D_i is the demand of the specific product variant i, and D_{tot} is the total demand of the products of the market space.

For this example, maximizing profit is the only objective. Although the augmented PPCTM can handle multiple objectives (as shown in Chapter 3), this example deals only with one objective in order to focus on the augmented method's ability to handle non-uniform demand. In the next example problem, the development of a process platform for customizable hearing aid shells (Chapter 5), the empirical validity of the use of the u-cDSP (Research Hypothesis #3) to handle multiple objectives will be tested.

4.2.3 Step 3: Identify the Modes for Managing Variety

In this step a designer must identify those modes of product design that are to be used to offer variety in volume and pressure. Overall, the purpose of the use of this method is to hierarchically organize these modes for managing product customization

and to select the best range of their commonality. After brainstorming, designers have chosen four modes of managing product variety - two for offering variety in volume, and two for offering variety in pressure. These modes are the same four that are presented in Hernandez's original work with the PPCTM (Hernandez, 2001).

Mode V1: Dimensional Customization of the Shell Length

One mode of managing variety in volume is to customize the shell length of each vessel. For this mode, changes in volume are achieved by cutting each shell length from a stock piece of raw material, while maintaining a constant radius. The result of this mode is the use of a common radius and a common stock length of material for a certain range of volumes. This mode is illustrated in Figure 4.5.

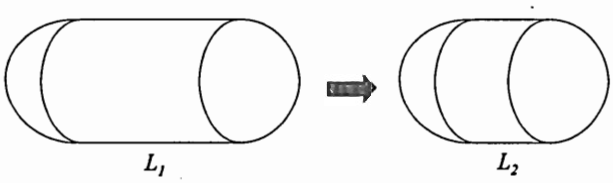


Figure 4.5 – Dimensional Customization of the Length of the Shell (Hernandez, 2001)

Mode V2: Modular Combination of Vessels

The other mode of managing variety in volume for this example problem is with the modular combination of the vessels. Shown in Figure 4.18, this mode is used to achieve discrete changes in volume by combining vessels of two lesser volumes. The result of the application of this mode is the determination of the range of volumes for which modularity is used.

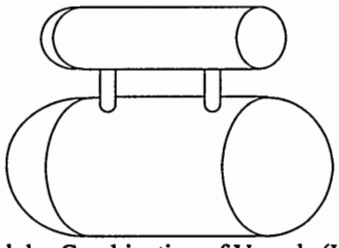


Figure 4.6 – Modular Combination of Vessels (Hernandez, 2001)

Mode P1: Standardization of the Shell and Head Thickness

This mode of managing variety in pressure involves the commonalization of the thickness of both the shell and the head of the vessel. It is based on the fact that if a shell with dimensions R and L, and thickness T_s and T_h satisfy the pressure constraints for a pressure P_1, it also satisfies these constraints for any pressure, $P \leq P_1$.

Mode P2: Commonalization of the Radius

This mode is based on the idea that changes in required pressure can also be satisfied by modifying the radius of a vessel. This mode requires the adjustment of the vessel length in order to maintain a constant volume. Of course, for each different radius used, a new press and die must be purchased. The result of the use of this mode is the determination of the range of pressures for which the radius will be commonalized.

These four modes (Modes V1, V2, P1, and P2) are the approaches considered for accessing all the points of the space of customization generated in Step 1 of this example. Next, in Steps 4 through 6, a hierarchic organization of these modes of managing variety is synthesized in order to offer a variety of products in the space of customization.

4.2.4 Step 4: Identify the Number of Hierarchy Levels and Allocate the Modes for Managing Variety to the Levels

Following the proposed augmented PPCTM, the modes for managing product variety are ordered in this step.

The First Stage and the First Space Element

As indicated in Section 2.1.2, the modes for managing product variety that are capable of achieving the smallest variations in design parameters are used first in the hierarchy. As such, for this first space element, modes V1 (dimensional customization of

shell length) and P1 (standardization of head and shell thickness) are used in concert to achieve variety in both volume and pressure.

Of the two modes for managing variety in volume, Mode V1 offers the highest fidelity changes in volume. By cutting the raw material, of length *Lo*, to any length desired, *L*, this mode is able to achieve continuous changes of volume from a volume V_2 to a volume $V \leq V_2$.

Small changes in the pressure parameter are achieved by changing the thickness of the shell and head of the vessel. This mode also offers an inexpensive way of offering variety in pressure. Mode P1, the commonalization of the thickness of the head and the shell, is used to define a pressure P_2 that can satisfy any pressure $P_1 \leq P_2$.

The size and shape of this first space element, S_1, is given by the value of the variables ΔV_1 and ΔP_1 as shown in Figure 4.7. ΔV_1 and ΔP_1 are therefore decision variables for this first stage.

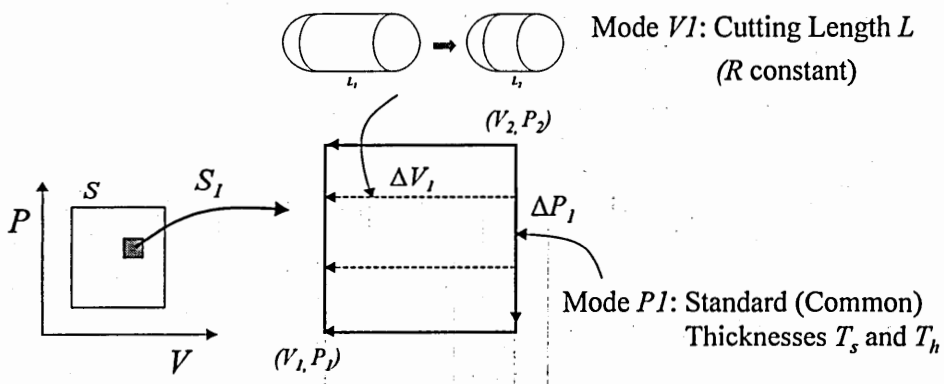


Figure 4.7 – The First Space Element of the Pressure Vessel Space for Customization (Hernandez, 2001)

The Second Stage and the Second Space Element

The second space element, S_2, is composed by a number of assemblies of the first space element, S_1, in the pressure dimension, as shown in Figure 4.8.

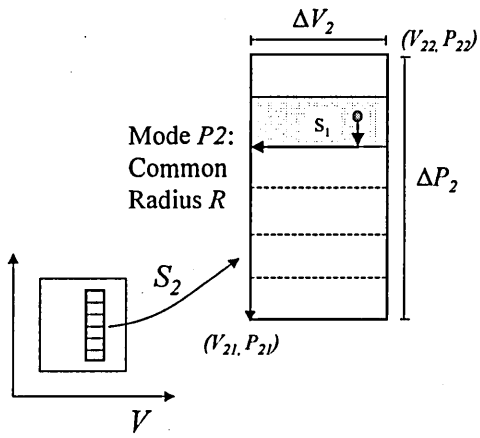


Figure 4.8 – The Second Space Element of the Pressure Vessel Space for Customization (Hernandez, 2001)

Only two modes for managing product variety are being considered for the pressure dimension. The first mode, Mode P1: Standardization of Head and Shell Thickness, is utilized in the synthesis of the first space element. Hence, there is only one mode of managing product customization to be used at this stage, namely, Mode P2: Commonalization of the Radius. Mode P2, changing the radius to achieve changes in pressure, is much more costly than Mode P1 due to the costs associated with purchasing the expensive manufacturing equipment.

Since the second space elements are composed of multiple first space elements in the P direction, the volume dimension of the second space elements, ΔV_2, are equivalent to the volume dimension of the first space elements, ΔV_1. The size and shape of this second space element is therefore given by the value of the variables ΔP_2 and ΔV_1 as shown in Figure 4.8.

<u>The Third Stage and the Third Space Element</u>

The third and final space element, S_3, is composed by a number of assemblies of the second space element, S_2, in the volume dimension, as shown in Figure 4.9.

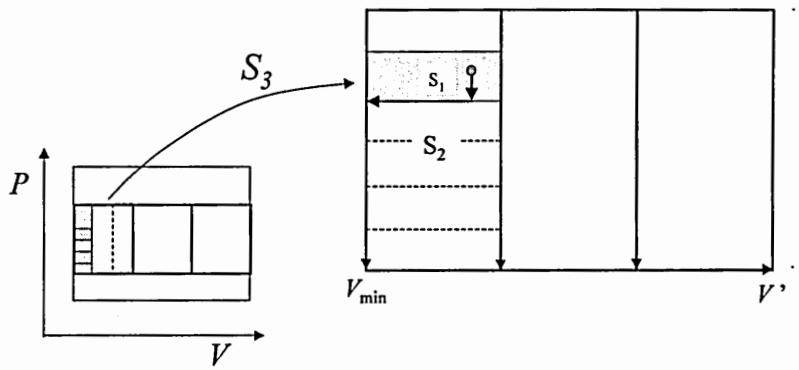


Figure 4.9 - The Third Space Element of the Pressure Vessel Example (Hernandez, 2001)

This space element utilizes the only mode of managing product customization that has not been used – Mode V2: Modular Combination of Vessels. The application of this mode is different than the application of the other modes in the previous stages. Mode V2 is based on the idea that large volume requirements can be satisfied by modularly combining two identical vessels of half the volume.

For example, if there exist customizable vessels in a range V_{min} to V', volumes larger than V' (up to $2V'$) can be achieved by combining vessels within this range. This is shown in Figure 4.33 where V' is 20 m^3. This modular combination limits the application of Mode V2 to discrete changes in volume. Since this mode can only achieve discrete changes in volume, it is used last in the hierarchy.

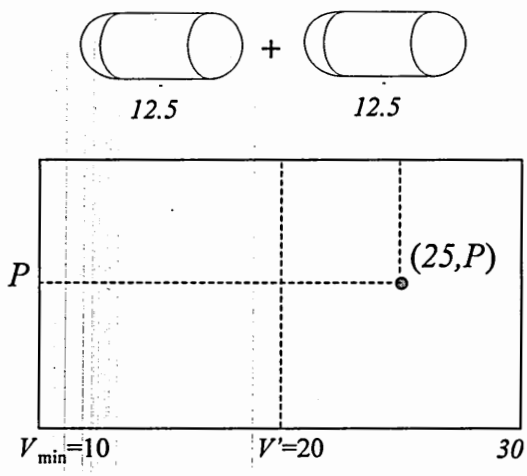


Figure 4.10– An Example of the Modular Combination of Vessels (Hernandez, 2001)

The decision associated with Mode V2 is the selection of "cut-off" point – a point where vessels with larger volumes are composed of two smaller vessels below the cut-off point (i.e., volume V' in the previous example).

A graphical interpretation of the synthesis of the modes for managing product variety, which results in a hierarchic association, is provided in Figure 4.11.

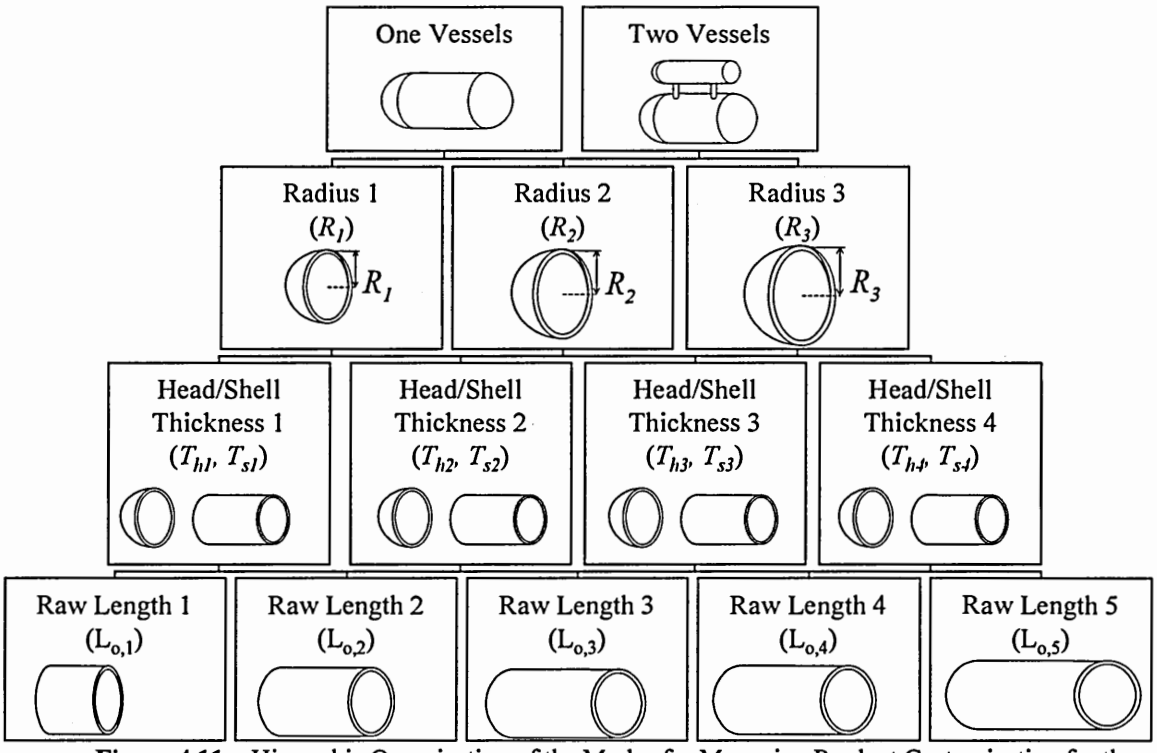


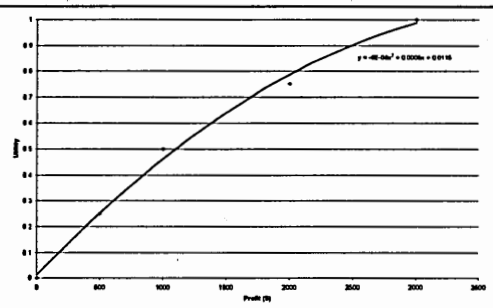
Figure 4.11 – Hierarchic Organization of the Modes for Managing Product Customization for the Pressure Vessel

4.2.5 Step 5: Formulate a Multi-Stage Utility-Based Compromise Decision Support Problem

There is only one objective in this example (i.e., maximizing profit), and there is no risk and uncertainty modeled in this specific problem. Thus, there is no specific need to formulate an expected utility function for the solution of the problem. Obviously, a designer executing this method for only one objective has a preference dedicated to improving this single objective. Thus, maximizing the expected utility of one objective is the same as improving that objective.

The formulation of the u-cDSP begins with the assessment of the expected utility function. After deliberation, a designer develops a utility function from the set of preferences shown in Table 4.1:

Table 4.1 – Designer Preferences for Profit of Pressure Vessel Example

Utility Value	*Profit ($)*	*Utility Function*	*Utility Plot*
1.0	3000	$U(AvgProf) = -6\text{x}10^{-8}\,AvgProf^2 + 0.0005AvgProf + 0.0115$ [4.16]	
0.5	2000		
0.25	1000		
0	0		

While the formulation of a multistage utility-based compromise Decision Support Problem may be too much for this example problem, it is used here in order to maintain consistency throughout this work.

<u>The First Stage and the First Space Element</u>

The first space element is defined by the range of application of Modes V1 and P1 (dimensional customization of shell length and standardization of shell and head thickness, respectively). These ranges of application are defined by the decision variables ΔV_1 and ΔP_1 (Section 4.2.4). Three design variables are commonalized in the first space element through the application of the two modes for managing customization, Mode V1 and Mode P1:

- the raw length, L_o
- the head thickness, T_h
- the shell thickness, T_s

The focus in this first decision stage is the selection of ΔV_1 and ΔP_1 to provide the largest average profit (Equation 4.15, Section 4.2.2). The average profit is determined by

calculating the difference between the selling price and the cost at each point in the space, and then averaging it over the entire space of customization.

The calculation of cost at each node begins with the determination of the thickness of the head and the shell. First, the largest pressure of the first space is defined as P_2, where (letting P_1 being the smallest pressure of the first space element):

$$P_2 = P_1 + \Delta P_1 \quad \text{(MPa)} \qquad [4.17]$$

From Mode P1, it is determined that if a vessel with dimensions R and L, and thickness T_s and T_h, satisfy the pressure constraints for a pressure P_2, it also satisfies these constraints for any pressure, $P_1 \leq P_2$. In the determination of the thickness of the head and shell, it is decided that a thickness that minimizes cost (thus maximizing profit) is simply the minimum thickness that satisfies the largest pressure P_2 given the radius R. From Equations 4.1 and 4.2, the thickness of the head and shell are calculated by:

$$T_s = \left(\frac{P_2}{\sigma_y - 0.6P_2} \right) R \qquad [4.18]$$

$$T_h = \left(\frac{P_2}{2\sigma_y - 0.2P_2} \right) R \qquad [4.19]$$

The focus in the application of Mode V1 is the commonalization of the raw length of the material, L_o. L_o is determined as the length required in making the largest volume of the first space, V_2.

$$L_0 = \frac{V_2}{\pi R^2} - \frac{4}{3} R \qquad [4.20]$$

where (letting V_1 being the smallest volume of the first space element):

$$V_2 = V_1 + \Delta V_1 \qquad [4.21]$$

The total material cost, welding cost, and waste cost of vessels within a single first space element, C_1, can now be calculated as the sum of all of the individual discrete vessels in the space, i (from Equation 4.10):

$$C_1 = \sum_{i=P_2-\Delta P_1}^{P_2} \sum_{j=V_2-\Delta V_1}^{V_2} \left[D(V_j, P_i) \cdot 2\pi\rho \begin{pmatrix} C_s R_{i,j} T_{s_{i,j}} L_{i,j} + C_h R_{i,j}^2 T_{h,i} + \frac{2}{9} C_w T_{s_{i,j}}{}^2 L_{i,j} + \\ \frac{4}{9} C_w \pi T_{s_{i,j}}{}^2 R_{i,j} + C_p T_{s_{i,j}} R_{i,j} \left(L_o - L_{i,j} \right) \end{pmatrix} \right] \quad [4.22]$$

where $D(V_j, P_i)$ is the number of vessels demanded for the specific variant (V_j, P_i), as determined by the demand function formulated in the first step of the method. The length of an individual vessel, $L_{i,j}$, is calculated by:

$$L_{i,j} = \frac{V_j}{\pi R^2} - \frac{4}{3} R_{i,j} \quad [4.23]$$

The total selling price of all the vessels of a single first space element, $SellingPrice_1$, is determined by summing the selling price of each individual vessel, i,j (from Equation 4.13):

$$SellingPrice_1 = \sum_{i=P_2-\Delta P_1}^{P_2} \sum_{j=V_2-\Delta V_1}^{V_2} \left[D(V_j, P_i) \cdot (C_v V_j + C_p P_i) \right] \quad [4.24]$$

The total profit of a first space element is evaluated as the difference between the cost and the selling price of each first space element (the difference between Equations 4.24 and 4.22).

$$Profit_1 = \left(SellingPrice_1 - C_1 \right) \quad [4.25]$$

After the ordering cost and the equipment cost are calculated, using Equations 4.11 and 4.12 respectfully, the total average profit of the entire space is determined by summing the profit of each first space element, s.

$$AvgProfit = \frac{1}{D_{tot}} \sum_{s}^{N_1} \left[\left(SellingPrice_{1,s} - C_{1,s} \right) - C_{order,s} - C_{equip,s} \right] \qquad [4.26]$$

where N_1 is the number of first space elements, S_1, in the entire space, S, and is calculated by:

$$N_1 = \frac{S}{S_1} = \frac{\Delta V \Delta P}{\Delta V_1 \Delta P_1} \qquad [4.27]$$

D_{tot} is the total demand of the entire space, calculated by:

$$D_{tot} = \sum_{i=P_{min}=10}^{P_{max}=20} \sum_{j=V_{min}=10}^{V_{max}=20} D(V_j, P_i) \qquad [4.28]$$

This variable is used to determine the average across all of the product variants of the entire space of customization.

In the formulation of the average profit, it is evident that there exists a tradeoff in the selection of the range of commonality of the raw length, L_o. From the calculation of waste cost, the last term of Equation 4.21, it is observed that the greater the difference between the raw length and the desired length of each vessel, $L_{i,j}$, the greater the cost. As a result, one would assume that there would be no incentive for commonalizing raw length; instead, it seems best to purchase raw lengths of metal at the exact length desired by the customer. The inclusion of the cost of ordering material (Equation 4.11), however, penalizes orders of multiple lengths of raw material because of shipping, handling, and stocking costs. The selection of ΔV_1 is therefore based on selecting the appropriate range of commonality in order to balance these two conflicting cost penalties.

As can be observed in Equation 4.26, the objective function is formulated in terms of three variables: ΔV_1, ΔP_1, and the radius, R. As discussed earlier, the radius is common across the first space element. This value of R, however, is not determined until

the decision stage using Mode P2 has been reached. As a result, these decisions are coupled.

The resulting decision formulation for this first stage is presented in Figure 4.12 (using the template from Section 3.4.5).

Decision 1: ΔV_1, ΔP_1

Given: The two-dimensional market space S = {(V, P)}
Mode V1: Dimensional Customization of the Shell Length
Mode P1: Standardization of the Head and Shell Thickness

Find: The value of decision variable ΔV_1, ΔP_1
The deviation variables, d_1^- and d_1^+

Satisfy: Bounds: $0 \le \Delta V_1 \le 20$
$0 \le \Delta P_1 \le 20$

Constraints: $T_s \ge \left(\frac{P}{\sigma_y - 0.6P} \right) R$

$T_h \ge \left(\frac{P}{2\sigma_y - 0.2P} \right) R$

$d_1^-, d_1^+ \ge 0$

$d_1^- \square d_1^+ = 0$

Goals: $E[u(AvgProf)] + d_1^- - d_1^+ = 1$

Minimize: $Z_1 = 1 - U_{AvgProf}$

where: AvgProf is calculated as in Equation 4.26, and
$U_{AvgProf}$ is calculated as in Equation 4.16, Table 4.1

Figure 4.12 – Formulation of the Multistage Compromise Decision Support Problem for the Pressure Vessel; First Stage Element

The Second Stage and the Second Space Element

The second space element is defined by the range of application of Mode P2: Commonalization of Radius. The size and shape of this second space element are defined by the value of the variables ΔP_2 and ΔV_1. The decision of ΔP_2 is akin to deciding the range for which the radius will be commonalized. As such, there are two

decision variables for this second stage, the range of radius commonality ΔP_2, and the value of the common radius of each second space element, R_j. Similar to the first stage, the focus of the second decision stage is the selection of a value of ΔP_2 that maximizes the average profit of the space of customization.

The cost of each second space element can be calculated as the sum of the cost of the first space elements contained within itself:

$$C_2 = \sum_{i=1}^{N_{1,2}} C_{1,i} \qquad [4.29]$$

where $C_{1,i}$ is the cost of each first space element (Equation 4.22), and $N_{1,2}$ represents the number of first space elements within a second space element:

$$N_{1,2} = \frac{S_2}{S_1} = \frac{\Delta P_2 \cdot \Delta V_2}{\Delta P_1 \cdot \Delta V_1} = \frac{\Delta P_2}{\Delta P_1} \qquad [4.30]$$

Similar to Equation 4.24, the total selling price of the vessels in the second space element is simply the sum of the products of the demand and the price of each vessel within the space. The average profit of the entire customization space is then calculated as:

$$AvgProfit = \frac{1}{D_{tot}} \sum_{i=1}^{N_2} \left[\left(SellingPrice_{2,i} - C_{2,i} \right) - C_{order,i} - C_{equip,i} \right] \qquad [4.31]$$

where N_2 is the number of second space elements, S_2, in the entire space, S, and is calculated by:

$$N_2 = \frac{S}{S_2} = \frac{\Delta V \Delta P}{\Delta V_2 \Delta P_2} \qquad [4.32]$$

D_{tot} is the total demand of the entire space and is calculated with Equation 4.28.

If the variables ΔV_1 and ΔP_1 are known, the choice of the radius for a specific second space element is given by the minimization of the cost of the space (Equation 4.29). It must be recalled from the first decision stage, however, that each radius is a function of ΔV_1 and ΔP_1, so these decisions are coupled.

The determination of each common radius, R_j, is based on the compromise between two inherent tradeoffs: the material cost, and the equipment cost. Ideally a designer would prescribe exactly each vessel's required radius in order to minimize material costs. This is not the most cost effective approach, however, as there is a significant equipment cost with the presses and the dies needed to manufacture each distinct radius. The selection of ΔP_2 and R_j are based on this inherent tradeoff.

The formulation of the decision of the second stage is provided in Figure 4.13.

<u>Decision 2: ΔP_2</u>

Given: The two-dimensional market space S = {(V, P)}
The value of ΔV_1, and ΔP_1 from the first stage
Mode P2: Commonalization of the Radius

Find: The value of decision variable ΔP_2
The deviation variables, d_2^- and d_2^+

Satisfy: Bounds: $0 \le \Delta P_2 \le 20$

Constraints: $\Delta P_1 \le \Delta P_2 \le 20$

$$d_2^-, d_2^+ \ge 0$$

$$d_2^- \square d_2^+ = 0$$

Goals: $E[u(AvgProf)] + d_2^- - d_2^+ = 1$

Minimize: $Z_2 = 1 - U_{AvgProf}$

where: AvgProf is calculated as in Equation 4.31, and
$U_{AvgProf}$ is calculated as in Equation 4.16, Table 4.1

Figure 4.13 – Formulation of the Multistage Compromise Decision Support Problem for the Pressure Vessel; Second Stage Element

The Third Stage and the Third Space Element

The third and final space element is defined by the range of application of Mode V2: Modular Combination of Vessels. The application of this mode is different than the application of the other modes in the previous stages. The decision associated with Mode V2 is the selection of "cut-off" point – a point where vessels with larger volumes are composed of two smaller vessels below the cut-off point. As such, the space element S_3 is composed of a number of spaces S_2 from the minimum volume of the space, V_{min}, to a cut-off point, V'. The pressure dimension for the third space is the same as the previous stage, (i.e., $\Delta P_3 = \Delta P_2$). The area of the third space element is calculated with:

$$S_3 = \Delta V_3 \Delta P_3 = (V' - V_{\min})\Delta P_2 \qquad [4.33]$$

As with the other stages, the focus of the implementation of Mode V2 is the maximization of profit. If no modular combination is attempted, i.e., $\Delta V_3 = 20$ (V'=30 m^3), the evaluation of the average profit will be exactly the same as the calculation of the average profit from the second stage (Equation 4.31):

$$AvgProfit = \frac{1}{D_{tot}} \sum_{i}^{N_2} \left[\left(SellingPrice_{2,i} - C_{2,i} \right) - C_{order,i} - C_{equip,i} \right] \qquad [4.34]$$

If however, a modular combination is attempted, the formulation of the average profit follows the same line of reasoning as previous stages: the cost is calculated by summing the cost of the smaller space elements that compose this space. The calculation of material cost, waste cost, and welding cost of each third space element is then:

$$C_3 = \sum_{j=1}^{N_{2,V'}} C_{2,j} + \sum_{j=1}^{N_{2,V'-V_{\max}}} 2C_{2,\frac{j}{2}} \qquad [4.35]$$

Where $N_{2,V'}$ is the number of second space elements between V_{min} and V'

$$N_{2,V'} = \frac{V' - V_{\min}}{\Delta V_2} \qquad [4.36]$$

and $N_{2,Vmax-V'}$ is the number of second space elements between V' and V_{max}

$$N_{2,V'-V\max} = \frac{V_{\max} - V'}{\Delta V_2} \qquad [4.37]$$

As can be observed from the Equation 4.35, the cost of vessels after the cut-off point, V', is double the cost of a vessel half its size. One must be careful in this analysis in the calculation of ordering cost and equipment cost for the third element; these costs are only calculated for the portion of the space before the cut-off point since only these vessels are directly manufactured. The average profit of the entire space with a $V' < V_{max}$ would then be:

$$\begin{aligned} AvgProfit = & \frac{1}{D_{tot}} \sum_{i=1}^{N_{2,V'}} \left[\left(SellingPrice_{2,i} - C_{2,i} \right) - C_{order,i} - C_{equip,i} \right] \\ & + \sum_{i=N_{2,V'}}^{N_{2,V'\max - V'}} 2 \left(SellingPrice_{2,\frac{i}{2}} - C_{2,\frac{i}{2}} \right) \end{aligned} \qquad [4.37]$$

The formulation of the decision of the third stage is provided in Figure 4.34.

A summary of the decision variables for each stage is provided in Table 4.2. The relationship between each stage is provided in Figure 4.15.

Table 4.2 – Decision Variables of Each Stage of Pressure Vessel Analysis

Stage	*Mode of Managing Customization*	*Decision Variable*	*Objective*
1	Mode V1 Mode P1	ΔV_1, ΔP_1	Maximize Average Profit, Equation 4.26
2	Mode P2	ΔP_2	Maximize Average Profit, Equation 4.31
3	Mode V2	ΔV_2	Maximize Average Profit, Equation 4.34 or 4.37

Decision 3: ΔV_2

Given: The two-dimensional market space S = {(V, P)}
The value of ΔV_1, ΔP_1, ΔP_2 from the first and second stages
Mode V2: Modular Combination of Vessels

Find: The value of decision variable ΔV_2
The deviation variables, d_3^- and d_3^+

Satisfy: Bounds: $0 \le \Delta V_2 \le 20$

Constraints: $\Delta V_1 \le \Delta V_2 \le 20$

$d_3^-, d_3^+ \ge 0$

$d_3^- \square d_3^+ = 0$

Goals: $E[u(AvgProf)] + d_3^- - d_3^+ = 1$

Minimize: $Z_3 = 1 - U_3$

where: $AvgProf$ is calculated as in Equation 4.34 and 4.37, and
$U_{AvgProf}$ is calculated as in Equation 4.16, Table 4.1

Figure 4.14 – Formulation of the Multistage Compromise Decision Support Problem for the Pressure Vessel; Third Stage Element

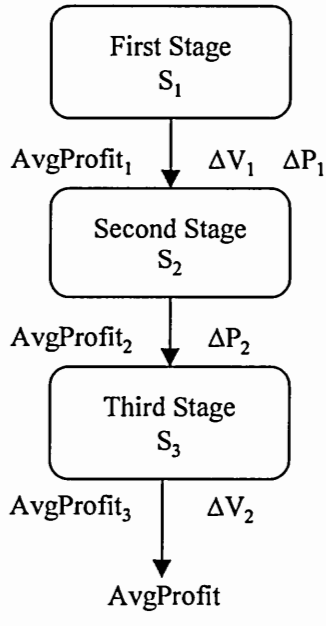


Figure 4.15 – Sequential Solution Process of the Three Stages

The sequential solution method illustrated in Figure 4.15 will not work for two reasons:

1) Each decision is coupled. For example, one cannot choose ΔV_1 and ΔP_1 without first knowing the commonalized radius that will be chosen in the second decision stage.
2) The optimal solution of one stage will not necessarily provide the global optimal solution. The selection of parameters that provide the most profit in the first space will not necessarily provide the most profit over the entire space.

As dictated by the augmented PPCTM presented in Chapter 3, these decisions need to be uncoupled and solved in a recursive manner. The result of the solution process will be the selection of the ranges of the modes of managing product customization that will produce the most profit for the manufacturing enterprise.

4.2.6 Step 6: Solve the Multi-Stage Utility-Based Compromise Decision Support Problem

The final step of this method is the formulation of an appropriate solution technique. As presented in Section 3.4.6, the solution of the multi-stage u-cDSP begins with the formulation of an appropriate solution algorithm for the determination of the ranges of managing modes for variety. Any appropriate solution algorithm can be used; its primary goal is the determination of the combination of the modes that provides the highest expected utility (i.e., maximum profit). A graphical representation of the solution method for this problem is presented in Figure 4.16 (following the general solution method presented in Figure 3.14).

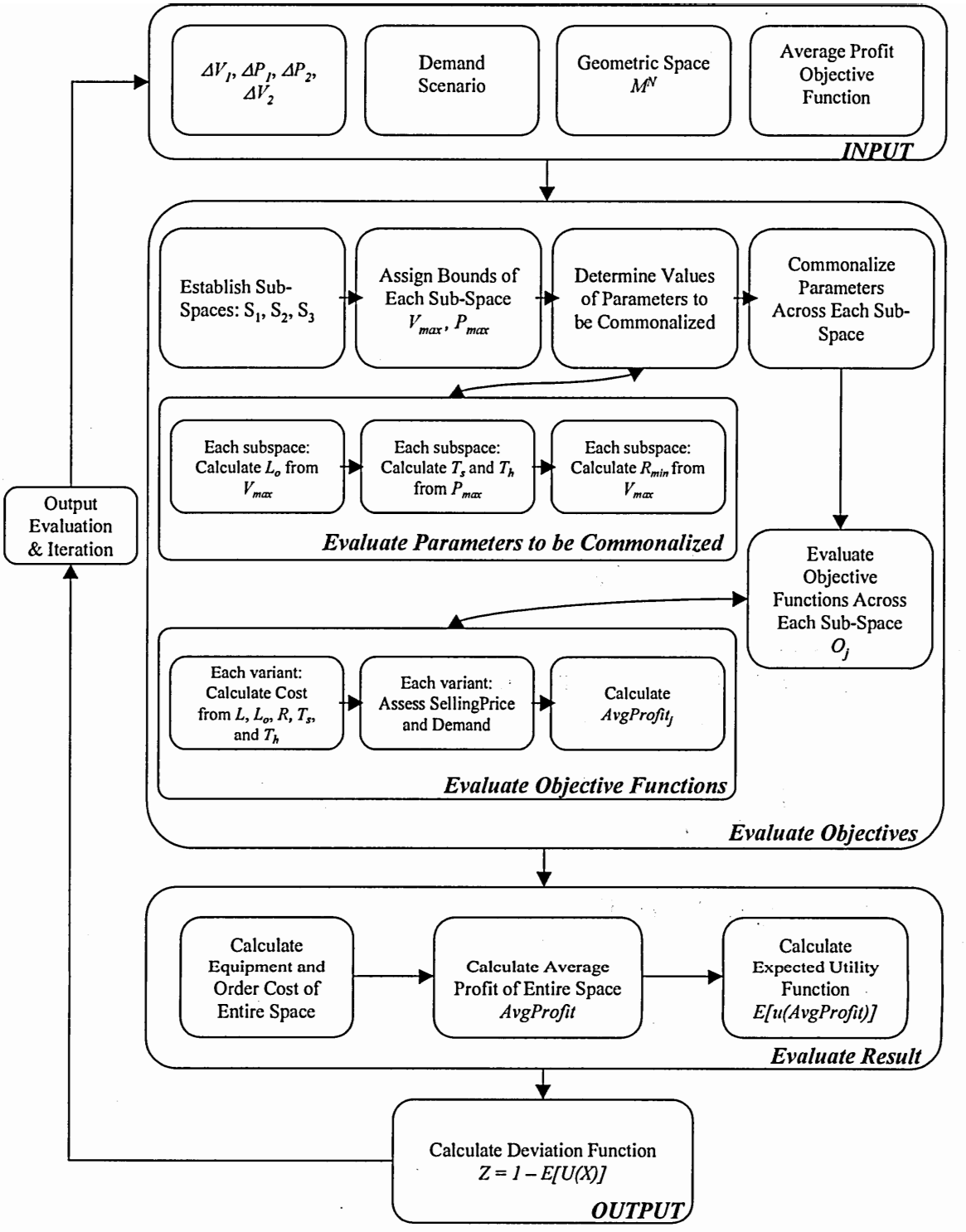


Figure 4.16 – Solution Algorithm for Pressure Vessel Example

As described in Section 3.4.6, this solution method involves iterating through values of the modes of managing variety (ΔV_1, ΔP_1, ΔP_2, and ΔV_2), establishing the dimensions of the sub-spaces, commonalizing the design parameters (L_o, T_s, T_h, and R) across each sub-space, evaluating the objective functions, and comparing the resulting overall utility of each iteration.

For this example problem it is noted that the determination of the values to commonalize for each mode for managing variety are determined through inherent design constraints (e.g., the value of raw length to commonalize is determined by the smallest feasible length for the maximum volume of the sub-space); as such, there is no need for an internal u-cDSP. This is illustrated in the following tutorial example.

Assume that the solution algorithm begins the search with the following initial ranges of application of the modes of managing product customization:

Table 4.3 – Example Values for Modes of Managing Product Customization

ΔV_1	5 m^3
ΔP_1	5 MPa
ΔP_2	10 MPa
ΔV_2	10 m^3

These values assist in the definition of each subspace. For this tutorial example, only the first subspace is considered. ΔV_1 and ΔP_1 compose the size of the first subspace (Section 4.2.4); therefore its bounds are identified by P_{max} = 15 MPa and V_{max} = 15 m^3 (Equations 4.17 and 4.21, respectively).

Within this space the raw length of material, L_o, and the head and shell thickness, T_h and T_s, are common. It is assumed that the common radius of this sub-space (from the second stage decision) is 1 m. The values of these parameters to be commonalized are determined by following the design constraints presented in Section 4.2.5. Specifically,

the shell will be produced from a plate with the minimum thickness that satisfies a pressure equal to P_{max} (15 MPa) when the radius is 1m. From Equation 4.18 this thickness is:

$$T_s = \frac{15\text{MPa}}{(1077\text{MPa} - 0.6(15\text{MPa}))}(1\text{m}) = 0.01404 \text{ m}$$

Similarly the head will be produced from a plate with thickness (Equation 4.19):

$$T_h = \frac{15\text{MPa}}{(2(1077\text{MPa}) - 0.2(15\text{MPa}))}(1\text{m}) = 0.0070 \text{ m}$$

The length of the raw plate is determined by the maximum volume of the subspace (15 m^3). The shell will be cut from a raw plate with length L_o (Equation 4.20):

$$L_0 = \frac{V_2}{\pi R^2} - \frac{4}{3}R = \frac{15\text{m}^3}{\pi(1\text{m})^2} - \frac{4}{3}(1\text{m}) = 3.44 \text{ m}$$

These parameters are then commonalized across the entire first sub-space.

To evaluate the cost of a single variant in this space, the length of a vessel L must first be determined (from Equation 4.23). The length of a vessel of $V = 12.5$ m^3 and $P =$ 11 MPa is:

$$L = \frac{V}{\pi R^2} - \frac{4}{3}R = \frac{12.5\text{m}}{\pi(1\text{m})^2} - \frac{4}{3}(1\text{m}) = 2.64 \text{ m}$$

With these parameter values obtained, a designer is able to calculate the cost of a specific variant across each first sub-space element (Equation 4.22). Coupled with the determination of the selling price function (Equation 4.24) and the demand function (Equation 4.28), the average profit of the first space element can be calculated (Equation 4.26).

With the general solution method outlined, the specifics of the solution method are explained in detail (Section 4.3). The results of the application of the augmented PPCTM are presented in Section 4.3.4. The validity of the results is discussed in Section 4.3.5.

4.3 SOLVING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE

The final step of the augmented PPCTM, as described in Section 3.4.6, is the solution of the multistage utility-based compromise Decision Support Problem. A general solution method for this example is presented in Section 4.2.6. Since the pressure vessel example is relatively simple, an exhaustive search of all possible combinations of values of ΔV_1, ΔP_1, ΔP_2, and ΔV_2 is performed (Section 4.3.2). In this section details of the solution method are presented. This is followed by a presentation and analysis of the results (Sections 4.3.4 and 4.3.5).

4.3.1 Discretization of the Geometric Space

There are two methods for analyzing a space of customization: through continuous evaluation, or through the discretization of the space (Section 3.4.2).

For a continuous evaluation, the objective function of each space is evaluated by integrating the function across the space. To do this, each objective function must be expressed as a function of the design parameters that are varied. For example, if using this technique for the pressure vessel example, the average profit must be expressed as a function of volume and pressure. While this represents the most mathematically rigorous solution technique, it is also the most complex. A series of derivations are required in order to express the objective function as a function of the design parameters. This limitation is especially relevant when dealing with non-uniform demand. While some

demand modeling methods easily approximate demand as a function of design parameters, the relationship between the demand of a variant and the design parameters of the variant can be difficult to approximate. This is especially evident when dealing with a demand scenario with discrete changes in demand, or when there is a random distribution of demand.

Recognizing the limitations of the continuous evaluation of the space of customization for this example problem, a discrete method of analysis is chosen for this example problem. The fundamental concept of a discrete solution technique is that an analysis of multiple discrete points of a space is an appropriate method of approximating the entire space. This analysis technique begins with a designer choosing an appropriate resolution for the spacing between each product variant. Nodes are then established across the space of customization. When stepping through the solution algorithm, the objective function is calculated at each node.

This method is appropriate for this problem because of the complex relationship between the objective function and the design parameters. Depending on the demand scenario, there is a complex relationship between demand and the dimensions of the space of customization. As mentioned above, this technique is the only manner in which one can model a random distribution of demand, as there exists no relationship between the design parameters and demand.

Some limitations of this method include:

- The numerical value of the range of modes for managing product customization must be a factor of the length of their respective dimension (i.e, ΔV must be a multiple of the range of volume variety offered by the manufacturer). Similarly,

the resolution (the spacing between nodes) must be a multiple of each of the ranges of the space dimensions.

- When first applying this method, a designer may not be confident in his/her selection of analysis resolution. Similar to the problems faced with finite-element analysis, one does not want to over-analyze the space (and thus not be efficient), nor does one want to under-analyze the space (and thus not get accurate results).
- The evaluation the objective function at each discrete node is computationally inefficient when compared to the evaluation of one objective function that integrates across the entire space in a continuous analysis.

As can be expected, the most apparent limitation of a discrete analysis is the number of calculations required to fully explore the design space. It is the author's opinion however, that this limitation is outweighed by the benefit of being able to analyze a wide variety of complex demand scenarios.

4.3.2 Performing an Exhaustive Search

As shown in Figure 4.17 (a view of the solution algorithm, Figure 4.16, from a more abstract level), the multistage decision presented in this problem is solved with an exhaustive search of the numerous combinations of the values of ΔV_1, ΔP_1, ΔP_2, and ΔV_2.

Each feasible combination of the ranges of the modes for managing variety is input into the solution algorithm. The value of the deviation function is returned from the analysis and is compared with previous output (the outer box of Figure 4.17). After each feasible combination is explored, the input that provides the smallest deviation from the goal is selected. The instantiation of this algorithm for the pressure vessel example problem is provided in Appendix A as a JAVA computer program.

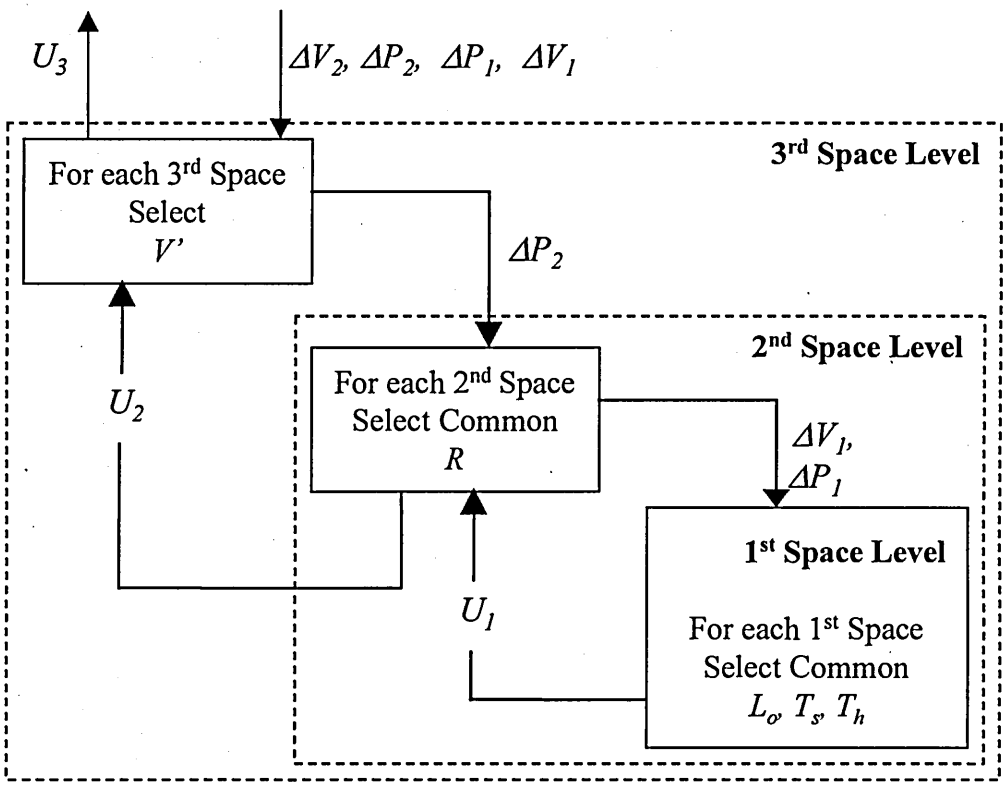


Figure 4.17 – Exhaustive Search of Design Parameters for the Pressure Vessel Example

The core benefit of the exhaustive search is the exploration of all combinations ensures that the solution reached is the best possible for this specific problem. Obviously, this rigorous method is not the most efficient, especially with a discretized space. While this approach is computationally expensive, it is of value for this specific example. Not only is the pressure vessel a relatively simple system to analyze, but also the number of possible combinations of the modes of managing customization is decreased because of the limitation that each mode must be a factor of the range of each design parameter dimension.

4.3.3 Modeling Demand

The first step of the augmented PPCTM involves the definition of the space of customization and its demand scenario. Three different methods of modeling demand are proposed in Section 4.2.1. In the solution of this problem for this thesis, ten different

demand scenarios were run by altering the parameters of each modeling technique (Table 4.4). This is done to prove that the formulation of the augmented PPCTM is independent of the way in which demand is modeled, and to articulate the lessons learned from observations gained from the results of the resultant product platforms. An acronym for each demand scenario title is provided to assist with the reference to specific results.

Table 4.4 – The Ten Demand Scenarios for the Pressure Vessel Example Problem

Demand Scenario	***Acronym***	***Reason for Inclusion***
Discrete Pyramid Large	DPL	Model discrete changes in demand
Discrete Pyramid Small	DPS	Model discrete changes in demand; model a market where there is no demand
Linear Large	LL	Model continuous change in demand
Linear Small	LS	Model continuous change in demand; observe effect of lower demand
Normal Center Small	NCS	Model Gaussian demand distribution
Normal High	NH	Model Gaussian demand distribution; observe effect of shifting mean of distribution
Normal Low	NL	Model Gaussian demand distribution; observe effect of shifting mean of distribution
Normal Center Large	NCL	Model Gaussian demand distribution; observe effect of increasing amount of demand
Random Large	RL	Model random distribution of demand
Random Small	RS	Model random distribution of demand; observe effect of lowering amount of demand

Demand Scenario #1, "Discrete Pyramid Large" (DPL)

This scenario is modeled for a space where there are large discrete changes in demand changes. The majority of the demand for the vessels is located at the center of the space. This demand scenario is illustrated in Figure 4.18.

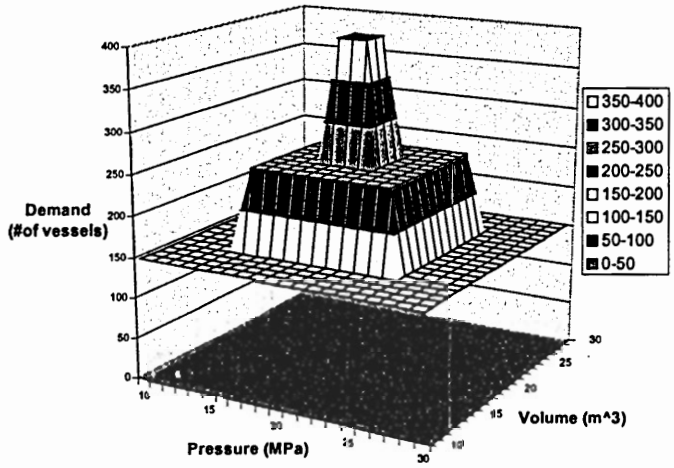


Figure 4.18 – Demand Scenario #1, "Discrete Pyramid Large"

Demand Scenario #2, "Discrete Pyramid Small" (DPS) – This scenario is very similar to the previous scenario. The key difference is that there exists a larger difference in customer preferences. This discrete is so large, in fact, that there are many vessels that have no demand at all. As such, this specific scenario tests the ability of the method to handle areas of zero demand. This scenario is shown in Figure 4.19.

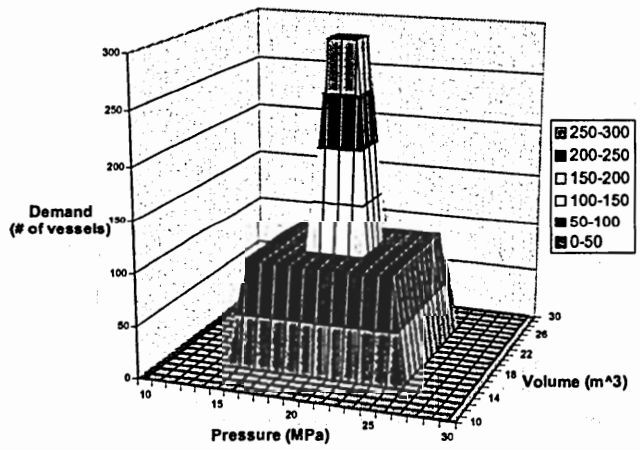


Figure 4.19 – Demand Scenario #2, "Discrete Pyramid Small"

Demand Scenario #3, "Linear Large" (LL) – This scenario uses the linear method of modeling demand. This scenario approximates a market where those higher-end vessels are of high demand. A graph of this scenario is presented in Figure 4.20.

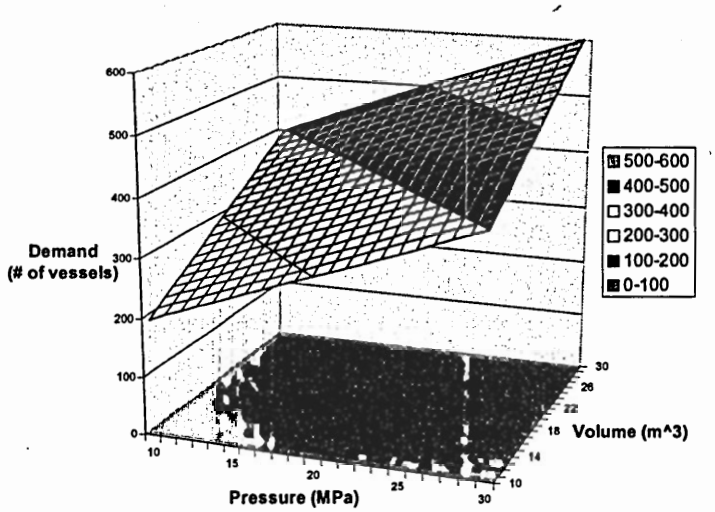


Figure 4.20 – Demand Scenario #3, "Linear Large"

Demand Scenario #4, "Linear Small" (LS) – This scenario is identical to the previous; however, the demand of each product is only a tenth of the demand of "Linear Large." This isobserved in Figure 4.21.

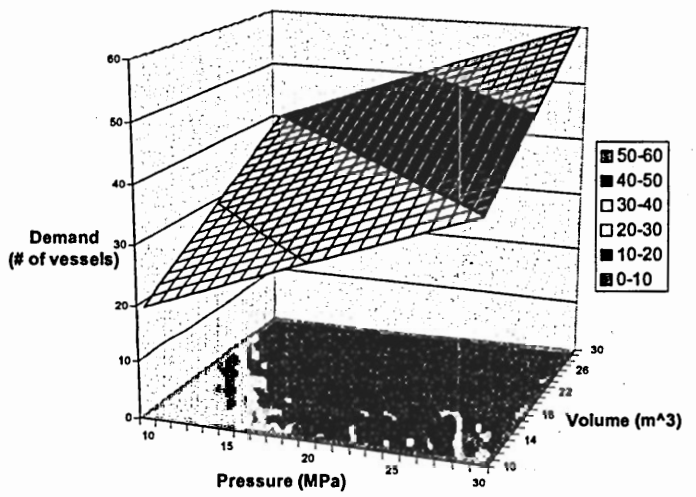


Figure 4.21 – Demand Scenario #4, "Linear Small"

Demand Scenario #5, "Normal Center Small" (NCS) – This scenario approximates the demand of the market as a Gaussian distribution (Section 4.2.1). This first normal approximation places the mean at the center of the space. This scenario is similar to "Discrete Pyramid Large," and is used as a basis for comparison.

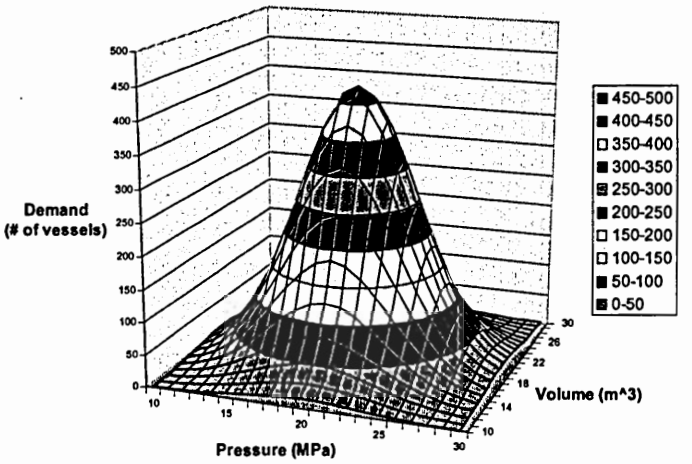


Figure 4.22 – Demand Scenario #5, "Normal Center Small"

Demand Scenario #6, "Normal High" (NH) – This scenario is similar to the previous demand scenario. Instead of placing the mean of the two-dimensional Gaussian distribution at the center of the market, this scenario approximates a market where the majority of the demand is for the lower-end vessels (small volume and small pressure requirements).

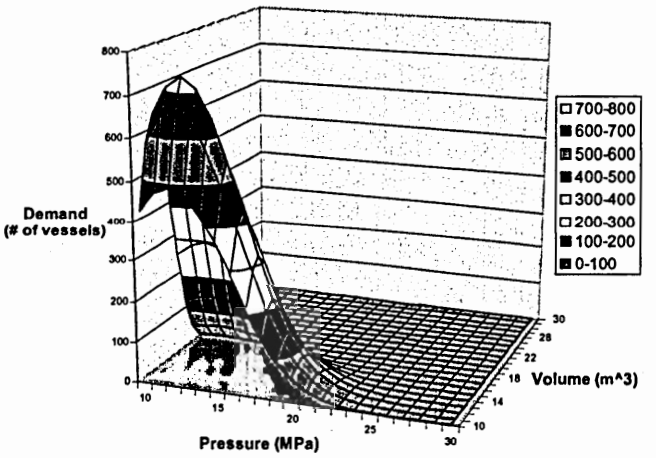


Figure 4.23 – Demand Scenario #6, "Normal High"

Demand Scenario #7, "Normal Low" (NL) – The seventh scenario is similar to the previous; however, a market where the majority of demand is centered on higher-end vessels is modeled.

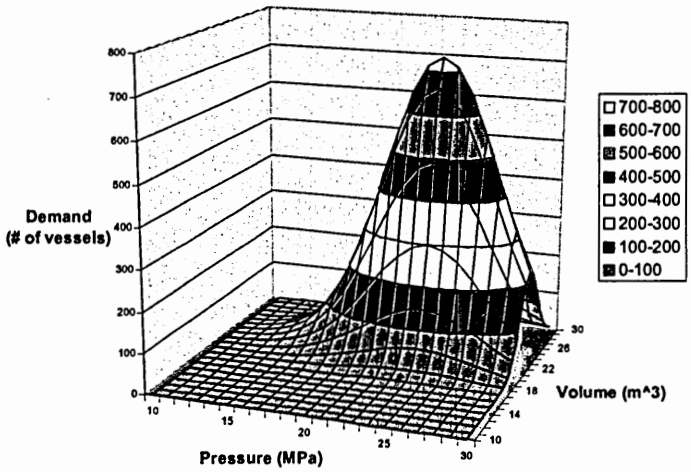


Figure 4.24 – Demand Scenario #7, "Normal Low"

Demand Scenario #8, "Normal Center Large" (NCL) – This demand scenario is the last use of the Gaussian approximation. It is similar to "Normal Center Large", where the majority of demand is located at the center of the customization space. The level of demand, however, is almost 150% larger than what is presented in Demand Scenario #5.

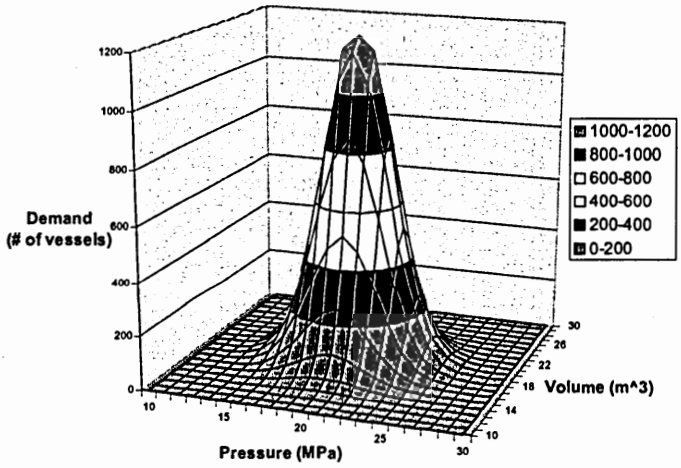


Figure 4.25 – Demand Scenario #8, "Normal Center Large"

Demand Scenario #9, "Random Large" (RL) – This last group of demand scenarios are of a modeling technique that has not been presented. Using a random number generator, the computer code programmed for this problem automatically generates a random distribution of demand (up to 500 vessels per variant). This modeling method is used to

represent a market where the demand for specific vessels does not show any type of reasonable trend. This is a technique that can only be analyzed by discretizing the customization space. It is hoped that this modeling technique can prove that the augmented PPCTM can handle any type of non-uniform demand distribution. The specific random distribution reported in this thesis is presented below.

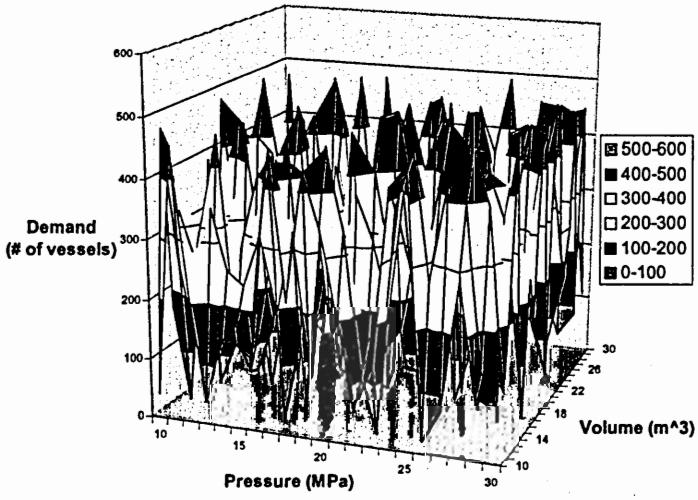


Figure 4.26 – Demand Scenario #9, "Random Large"

Demand Scenario #10, "Random Small" (RS) – This final demand scenario is identical to the previous scenario in the fact that it is also developed with a random number generator. The key difference lies in the fact that this distribution only allows a maximum demand of 10 vessels per variant. This scenario will produce an opportunity to make observations regarding validation.

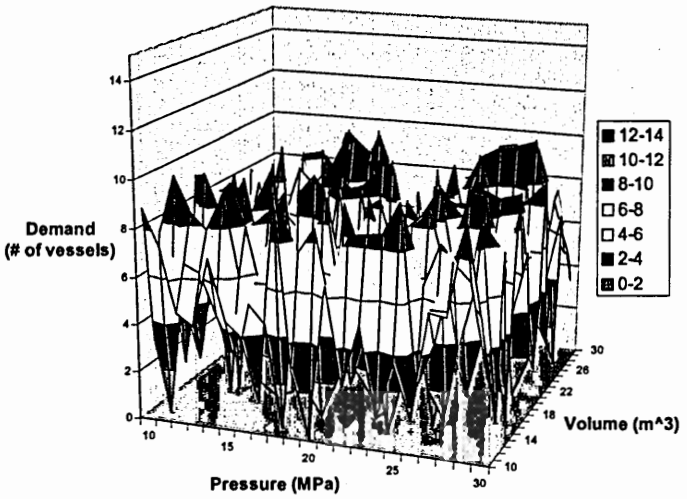


Figure 4.27 – Demand Scenario #10, "Random Small"

With the specific details of the solution method presented, the results of the application of the augmented PPCTM to each demand scenario of this example are presented in Section 4.3.4.

4.3.4 Evaluating the Results

As indicated in Section 4.3.1, the application of the augmented PPCTM to the pressure vessel example is carried out for ten different non-uniform demand scenarios. Identified by acronym, the results of each of these scenarios are presented in this section. Before a presentation of the full range of results, the results of the first demand scenario are provided as a tutorial on how to interpret the results.

Demand Scenario #1, "Discrete Pyramid Large" (DPL)

The first demand scenario is plotted in Figure 4.18. It models a market where the majority of demand is centered in the market and large, discrete differences in the demand of other vessels are present in the space of customization. The results of the application of the augmented PPCTM to this demand scenario are presented in Table 4.5:

Table 4.5 – Results for Demand Scenario #1, "Discrete Pyramid Large"

ΔV_1 (m^3)	ΔP_1 (MPa)	ΔP_2 (MPa)	ΔV_3 (m^3)	Average Profit ($)
2	1	20	20	1998.64

These results inform the manufacturing enterprise that the best configuration of the modes of managing customization to maximize profit is to commonalize the raw length for every 2 m^3 of volume, commonalize the head and shell thickness for every 1 MPa of pressure, commonalize the radius of the vessel for every 20 MPa, and to not modularly combine vessels. This space of customization is presented in Figure 4.28.

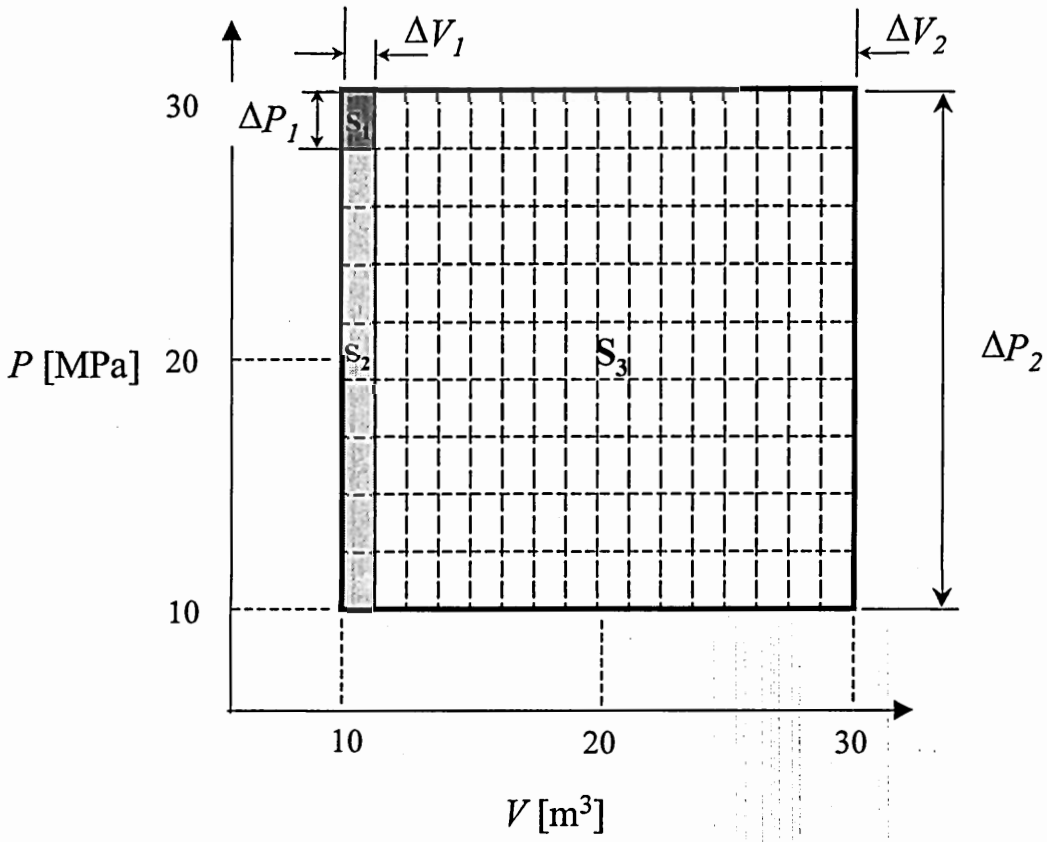


Figure 4.28 – Resulting Space of Customization for Demand Scenario #1, "Discrete Pyramid Large"

The specific values of the design variables are also provided by the augmented PPCTM. These values are presented in Table 4.6.

Table 4.6 – Mapping Between Product Specifications and Design Variables for Demand Scenario #1, "Discrete Pyramid Large"

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		6.57	3.27	
	11		7.24	3.60	
	12		7.90	3.93	
	13		8.56	4.25	
	14		9.22	4.58	
	15		9.89	4.91	
	16		10.55	5.24	
	17		11.22	5.57	
	18		11.89	5.89	
	19		12.55	6.22	
10	20	0.704	13.22	6.55	5.482
	21		13.89	6.88	
	22		14.56	7.21	
	23		15.23	7.53	
	24		15.90	7.86	
	25		16.57	8.19	
	26		17.25	8.52	
	27		17.92	8.85	
	28		18.60	9.18	
	29		19.27	9.51	
	30		19.95	9.83	
	10		6.57	3.27	
	11		7.24	3.60	
	12		7.90	3.93	
	13		8.56	4.25	
	14		9.22	4.58	
	15		9.89	4.91	
	16		10.55	5.24	
	17		11.22	5.57	
	18		11.89	5.89	
	19		12.55	6.22	
12	20	0.704	13.22	6.55	6.766
	21		13.89	6.88	
	22		14.56	7.21	
	23		15.23	7.53	
	24		15.90	7.86	
	25		16.57	8.19	
	26		17.25	8.52	
	27		17.92	8.85	
	28		18.60	9.18	
	29		19.27	9.51	
	30		19.95	9.83	
	10		7.08	3.52	
	11		7.79	3.87	
	12		8.50	4.23	
	13		9.21	4.58	
	14		9.93	4.93	
	15		10.64	5.28	
	16		11.36	5.64	
	17		12.08	5.99	
	18		12.79	6.34	
	19		13.51	6.70	
14	20	0.758	14.23	7.05	6.750
	21		14.95	7.40	
	22		15.67	7.76	
	23		16.39	8.11	
	24		17.12	8.46	
	25		17.84	8.82	
	26		18.56	9.17	
	27		19.29	9.52	
	28		20.01	9.88	
	29		20.74	10.23	
	30		21.47	10.58	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.54	3.75	
	11		8.30	4.13	
	12		9.06	4.50	
	13		9.82	4.88	
	14		10.58	5.26	
	15		11.34	5.63	
	16		12.11	6.01	
	17		12.87	6.38	
	18		13.63	6.76	
	19		14.40	7.14	
16	20	0.808	15.17	7.51	6.732
	21		15.93	7.89	
	22		16.70	8.27	
	23		17.47	8.64	
	24		18.24	9.02	
	25		19.01	9.40	
	26		19.78	9.77	
	27		20.56	10.15	
	28		21.33	10.53	
	29		22.10	10.90	
	30		22.88	11.28	
	10		7.97	3.97	
	11		8.78	4.37	
	12		9.58	4.76	
	13		10.38	5.16	
	14		11.19	5.56	
	15		11.99	5.96	
	16		12.80	6.35	
	17		13.61	6.75	
	18		14.42	7.15	
	19		15.23	7.55	
18	20	0.854	16.04	7.94	6.717
	21		16.85	8.34	
	22		17.66	8.74	
	23		18.47	9.14	
	24		19.29	9.54	
	25		20.10	9.93	
	26		20.92	10.33	
	27		21.74	10.73	
	28		22.55	11.13	
	29		23.37	11.53	
	30		24.19	11.93	
	10		8.38	4.17	
	11		9.23	4.59	
	12		10.07	5.01	
	13		10.92	5.43	
	14		11.76	5.84	
	15		12.61	6.26	
	16		13.46	6.68	
	17		14.31	7.10	
	18		15.16	7.52	
	19		16.01	7.93	
20	20	0.898	16.86	8.35	6.699
	21		17.72	8.77	
	22		18.57	9.19	
	23		19.42	9.61	
	24		20.28	10.03	
	25		21.14	10.45	
	26		21.99	10.86	
	27		22.85	11.28	
	28		23.71	11.70	
	29		24.57	12.12	
	30		25.44	12.54	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.77	4.36	
	11		9.65	4.80	
	12		10.54	5.24	
	13		11.42	5.68	
	14		12.30	6.11	
	15		13.19	6.55	
	16		14.08	6.99	
	17		14.97	7.42	
	18		15.86	7.86	
	19		16.75	8.30	
22	20	0.939	17.64	8.74	6.687
	21		18.53	9.17	
	22		19.42	9.61	
	23		20.32	10.05	
	24		21.21	10.49	
	25		22.11	10.93	
	26		23.01	11.36	
	27		23.90	11.80	
	28		24.80	12.24	
	29		25.70	12.68	
	30		26.61	13.12	
	10		9.14	4.55	
	11		10.06	5.00	
	12		10.98	5.46	
	13		11.90	5.91	
	14		12.82	6.37	
	15		13.74	6.82	
	16		14.67	7.28	
	17		15.59	7.73	
	18		16.52	8.19	
	19		17.45	8.65	
24	20	0.979	18.38	9.10	6.674
	21		19.31	9.56	
	22		20.24	10.01	
	23		21.17	10.47	
	24		22.10	10.93	
	25		23.03	11.38	
	26		23.97	11.84	
	27		24.91	12.30	
	28		25.84	12.75	
	29		26.78	13.21	
	30		27.72	13.67	
	10		9.49	4.72	
	11		10.44	5.19	
	12		11.40	5.67	
	13		12.35	6.14	
	14		13.31	6.61	
	15		14.27	7.09	
	16		15.23	7.56	
	17		16.19	8.03	
	18		17.15	8.50	
	19		18.12	8.98	
26	20	1.016	19.08	9.45	6.663
	21		20.05	9.92	
	22		21.01	10.40	
	23		21.98	10.87	
	24		22.95	11.35	
	25		23.92	11.82	
	26		24.89	12.29	
	27		25.86	12.77	
	28		26.83	13.24	
	29		27.81	13.72	
	30		28.78	14.19	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		9.82	4.89	
	11		10.81	5.38	
	12		11.80	5.87	
	13		12.79	6.36	
	14		13.78	6.85	
	15		14.78	7.34	
	16		15.77	7.83	
	17		16.76	8.32	
	18		17.76	8.81	
	19		18.76	9.30	
28	20	1.052	19.76	9.79	6.651
	21		20.76	10.28	
	22		21.76	10.77	
	23		22.76	11.26	
	24		23.76	11.75	
	25		24.76	12.24	
	26		25.77	12.73	
	27		26.78	13.22	
	28		27.78	13.71	
	29		28.79	14.20	
	30		29.80	14.69	
	10		10.15	5.05	
	11		11.17	5.56	
	12		12.19	6.06	
	13		13.21	6.57	
	14		14.24	7.07	
	15		15.26	7.58	
	16		16.29	8.08	
	17		17.32	8.59	
	18		18.35	9.10	
	19		19.38	9.60	
30	20	1.087	20.41	10.11	6.637
	21		21.44	10.62	
	22		22.47	11.12	
	23		23.51	11.63	
	24		24.54	12.14	
	25		25.58	12.64	
	26		26.62	13.15	
	27		27.66	13.66	
	28		28.70	14.16	
	29		29.74	14.67	
	30		30.78	15.18	

Table 4.6 serves as a "roadmap" for the designer; any potential combination of pressure and volume for the considered space of customization is connected to specific design parameters, including dimensions of raw material.

In order to explain this table the following tutorial example is provided: imagine a customer that requests a pressure vessel with volume of 17.25 m^3 and a pressure of 11.5 MPa. Looking to Table 4.6, the nearest volume above that which is requested is 18 m^3.

The second column of the table shows that the closest pressure to that requested is 12 MPa. From the corresponding row, the resulting vessel would have a radius 0.854 m, a shell thickness of 9.58 mm, and a head thickness of 4.76 mm. The length of the specific vessel requested, calculated from Equation 4.23, is:

$$L = \frac{V}{\pi R^2} - \frac{4}{3}R = 6.39 \text{ m}$$

This length will be cut from the raw material of length (from the last column of the above table) 6.72 m.

Thus, not only does the use of the augmented PPCTM identify the most suitable ranges of commonality, but its use also provides a designer with a roadmap of design variables. As a side note, had ΔV_3 been less than 20 m^3, any vessel greater than the identified cut-off point would have been made from two vessels of half the volume.

In an effort to maintain brevity, a summary of the results from each demand scenario is presented in Table 4.7. The specific mapping of the product specifications to design variables, as shown in Table 4.5 for the first demand scenario, are presented for each demand scenario in Appendix B.

Table 4.7 – Range of Commonality of the Modes for Managing Product Customization for each Demand Scenario of the Pressure Vessel Example

Demand Scenario #	Demand Scenario Acronym	ΔV_1 (m^3)	ΔP_1 (MPa)	ΔP_2 (MPa)	ΔV_2 (m^3)	Average Profit ($)
1	DPL	2	1	20	20	1998.64
2	DPS	5	1	20	15	1878.34
3	LL	1	1	20	20	1876.99
4	LS	4	1	20	20	1622.26
5	NCS	2	1	20	20	1976.05
6	NH	2	1	20	20	1041.32
7	NL	2	1	20	12	2519.62
8	NCL	2	1	20	16	2019.26
9	RL	1	1	20	20	2007.10
10	RS	10	1	20	20	1118.60

The validity of these results in analyzed in Section 4.3.5.

4.3.5 Analysis of the Results

In this section, the results are analyzed in three separate steps:

- establishing the validity of the results,
- exploring the solution algorithm, and
- articulating lessons learned from applying the augmented PPCTM to different demand scenarios.

Establishing the Validity of the Results

Before the articulation of reflections of the results is presented, the validity of the results must first be presented. The validity of the results is presented through two exercises; namely through checking the constraints for a portion of the exhaustive search history, and through the comparison of results of similar demand scenarios.

In order to check that the constraints of the multi-stage u-cDSP are valid, a small snapshot of the history of the exhaustive search solution algorithm is shown for the "Linear Large" demand scenario (Demand Scenario #3) in Table 4.8.

Table 4.8 – Checking Constraints for Pressure Vessel Example

Iteration #	*Δ V1*	*Δ P1*	*Δ P2*	*Δ V3*	*Profit ($)*	*U*	*d-*	*d+*
1	1	1	20	20	1876.99	0.7386	0.2614	0
10	1	2	12	12	573.98	0.2787	0.7213	0
22	1	2	2	18	1415.25	0.5989	0.4011	0
48	1	1	14	11	629.48	0.3025	0.6975	0
60	2	4	14	18	1046.73	0.4691	0.5309	0
73	2	5	15	18	880.43	0.4052	0.5948	0
85	4	2	12	16	930.97	0.4250	0.5750	0
120	4	1	14	12	557.22	0.2715	0.7285	0
143	4	2	14	12	361.77	0.1845	0.8155	0
147	4	2	14	16	930.97	0.4250	0.5750	0
160	5	1	10	10	213.12	0.1153	0.8847	0
175	10	1	1	20	1457.11	0.6127	0.3873	0

From Table 4.8 it is observed that the two major constraints of each u-cDSP are met:

- the range of application of each mode is smaller than those modes that are applied at higher levels in the hierarchy (i.e., $\Delta r_i \leq \Delta r_{i+1}$)
- $d_i^-, d_i^+ \geq 0$ and $d_i^- \square d_i^+ = 0$. As expected, the dot product of the deviation variables is always equal to zero. The formulation of the expected utility function insures that this constraint will be met due to the inability of the system to meet a designer's "ideal" preference value (i.e., the system will never "over-achieve" the goal).

Another manner in which the validity of the results is checked is through the comparison of "Discrete Pyramid Large" (Figure 4.18) and "Normal Center Small" (Figure 4.22) demand scenarios. Both demand scenarios approximate a market where the majority of demand is located in the center of the market space. Despite the opposing adjectives in their names, both scenarios have the same amount of total demand in the market space. Comparing the results reported in Table 4.7, one observes that the selection of the ranges of each mode of managing product customization are exactly the same. Their corresponding values of maximum profit are also roughly the same (a difference of 1.13%). The fact that different demand scenarios approximating similar market conditions produce similar results provides proof that the augmented PPCTM can successfully handle a wide variety of non-uniform demand scenarios.

Solution Algorithm Observations

From further exploration of the results presented in Table 4.7, a series of observations on the solution algorithm can be identified:

- ΔP_1 (range of Mode P1, standardization of the head and shell thickness) is always equal to 1 MPa,
- ΔP_2 (range of Mode P2, commonalization of the radius) is always equal to 20 MPa, and
- ΔV_2 (modular combination of vessels) is almost always 20 m^3

These results are described in detail below.

ΔP_1 is always equal to 1 MPa: At first glance this seems like a major error. Through deeper analysis of the solution method, the reason for this occurrence is apparent. For most modes for managing product variety (Section 4.2.4) there is a reason for commonalizing the design parameter due to inherent tradeoffs in cost (i.e., commonalizing the radius increases material cost, but drives down equipment cost). Mode P1 is the only mode in which there is no such cost tradeoff; no money is saved through commonalization. As such, it is logical that the solution algorithm would never choose a ΔP_1 greater than the set discretizing resolution; doing so would make vessels have larger shell and head thickness than what is required, thus increasing material cost. In other words, if there is no benefit or discount obtained by commonalizing the thickness, ΔP_1 will always be equal to the smallest resolution available. In retrospect, some sort of discount should have been accounted for (a discount in purchasing raw material of the same thickness). As presented here, however, this provides a manner in which to verify the logic of the method.

ΔP_2 is always equal to 20 MPa: The reason why this occurs is similar to the explanation of the last problem. As one can recall, there is a tradeoff for the commonalization of the radius: prescribing fewer radii will increase the average material

cost while decreasing the equipment cost. This tradeoff, however, does not occur during this particular subspace. The reason for commonalizing the radius in this subspace is to have another method of achieving distinct changes in pressure (through the calculation of the head and shell thickness, Equations 4.1 and 4.2). The resultant material cost does not provide enough leverage against the heavy equipment cost. As such, the radius is actually commonalized during the implementation of the first space: the largest volume in the first space prescribes a minimum radius for that space. As a result, the combination of first spaces into a second space will result in the use of the same radius across the entire space. Therefore, for any demand scenario the radius will always be commonalized across each second space.

ΔV_2 is almost always 20 m^3: The reason for repeated results in the above observation was that the equipment costs were too large to choose anything except commonalization across the space. The problem in this observation is the opposite: the material costs are not high enough to compensate for large material costs. When a cutoff point is established through the use of Mode V2, the material cost of the vessels after this point is calculated as twice the cost of a vessel of half its volume. This, upon further inspection, is a large penalty. Consider a vessel of 22 m^3 and 10 MPa which, as a single vessel, costs ~$3275.08 to manufacture. If this specific vessel is made by combining two vessels of 11 m^3, the manufacturing cost is $4976.26 (2 x $2488.13), a difference of $1701.18. For many demand scenarios, the resulting increase in material cost by modularly combining vessels was too much to compensate the decrease in the purchase cost of an additional radius press. Specifically the scenarios in which $\Delta V_3 < 20$ were

those scenarios in which the amount of demand for the higher-end vessels was very small.

These results provide insight into the limitations of the modeling of the pressure vessel problem. While more care should be taken into the modeling of specific tradeoffs between material costs and ordering costs, one must recognize the value of realizing that commonalization, for some modes, is not an appropriate avenue to pursue. While the lack of complexity of the pressure vessel may be considered a limitation of this work, it should not be considered a weakness of the augmented PPCTM. The application of the augmented method has demonstrated that its use can assist a designer in the identification of the proper ranges of commonality in each mode of managing product customization.

Lessons of Product Platform Design

While the limitations of the simple pressure vessel model prevent the ability for the discovery of new and exciting epiphanies in product platform design, some lessons can be identified.

- low demand leads to more commonality,
- large demand leads to more variety,
- product platform design is a good approach to offering variety affordably, and
- commonality is not profitable when demand is large and extremely non-uniform.

These lessons are presented in detail below.

Low demand leads to more commonality: When comparing the "large" and "small" variations of each modeling method, one observes that the range of commonality for each mode of managing product customization increases as the total amount of demand is

decreased (i.e., DPL vs. DPS, LL vs. LS, and RL vs. RS in Table 4.7). For a manufacturing enterprise, this makes sense. Lower demand translates into lower total profit. This lower amount of incoming revenue makes it much harder to cover the extra cost of offering variety. In this example, lower demand prevents the purchase of multiple presses and dies for different radii.

Large demand leads to more variety. The observation of this lesson comes from the flipside of the previous observation. Larger demands translate to larger profits. As such, a manufacturer can afford to offer numerous radii. It is inherently assumed in this example that the manufacturer wishes to meet customers' demand as well as possible; it is possible that the customers could go to a competitor if there needs are not met. As a result, a manufacturer will always choose to offer as much variety as can be allowed.

Product platform design is a good approach to offering variety affordably: The concept of mass production presents two options for a manufacturer when faced with this specific problem: (i) make one type of vessel, the largest of the market space, to satisfy the needs for each different demand ($\Delta V_1=\Delta P_1=\Delta P_2=\Delta V_2=20$), and (ii) make one vessel for every specific demand ($\Delta V_1=\Delta P_1=\Delta P_2=1/\infty$). As can be seen from the results presented, neither of these solutions to the problem provides the most profit. The only exception to this is observation is demand scenarios LL and RL that, in turn, provide another lesson learned.

Commonality is not profitable when demand is large and extremely non-uniform: There are only two demand scenarios for which no amount of commonality proved to be the most profitable. These two scenarios, "Linear Large" (Figure 4.20) and "Random Large" (Figure 4.26), model demand differently; however, these two specific models

approximate a similar demand scenario, namely a market with a large amount of demand scattered about the space. When such a large demand is scattered about the space, there is no reason to commonalize any design parameter; the large demand would provide enough profit to cover large equipment costs, and its even distribution across the space would give no reason to commonalize certain parameters.

4.4 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD

The reasons for, and a description of, the augmented PPCTM have been presented in the first three chapters of this thesis. In this fourth chapter, the augmented PCPTM is applied to an example problem involving the product platform design for a line of customizable pressure vessels. In this chapter the empirical structural and empirical theoretical validation of the presented augmented PPCTM is established (Figure 4.29).

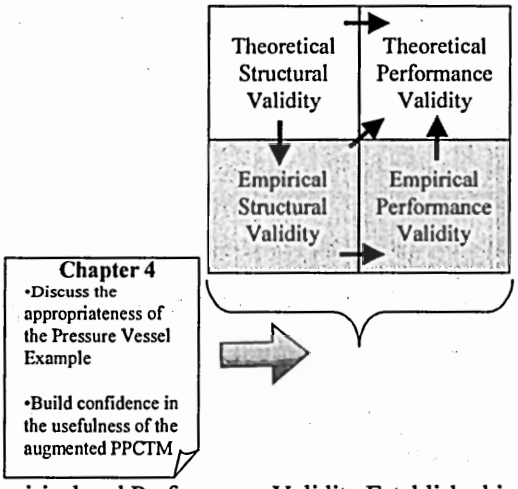


Figure 4.29 – Empirical and Performance Validity Established in Chapter 4

As discussed in Section 1.4.2, the establishment of empirical structural validation involves building confidence in the appropriateness of the example problem for illustrating and verifying the performance of the design method. Pressure vessel design is a classic mechanical engineering design example. The simplicity of its analysis provides

an opportunity for an engineer to focus on the design method at hand without having to worry about a complex analysis. While the modeling of the vessel is relatively simple, the selection of appropriate design variables can be a difficult task due to the mix of continuous and discrete variables and non-linear functions and models.

The selection of this specific example problem is also especially relevant as it is the same problem that has been analyzed using Hernandez's original PPCTM (Hernandez, 2001). As such, direct comparisons between the two methods can be drawn. If the original PPCTM is assumed to be a valid method, then the use of a similar example problem to test an augmented PPCTM is also valid.

As presented in Section 1.4.3, empirical performance validity involves building confidence in the usefulness of the method (as far as the degree to which the method's purpose has been achieved) using example problems. A presentation and critical analysis of the results of the application of the augmented PPCTM is discussed in Section 4.3.5. In this analysis, several points regarding the validity of the results are raised and then subsequently explained.

The purpose of this specific application of the augmented PPCTM is to show that the augmented PPCTM can be applied to a market with of any type of non-uniform demand. Ten different demand scenarios are run in order to prove that the augmented PPCTM could handle a wide variety of demand modeling techniques (Section 4.3.4). Each scenario provides logical and valid results (Section 4.3.5).

In the Chapter 5, empirical structural and empirical performance validity of the augmented PPCTM is further examined with a second example problem, namely, the design of a process platform for a line of customizable hearing aid shells. This example

problem will test the validity of the application of the augmented PPCTM to process platform development. This problem is inherently more complex than the pressure vessel example problem. Modeling the rapid production process of a line of customizable hearing aid shells provides many more design parameters and multiple objectives. This problem provides an opportunity to test the ability of the augmented PPCTM to handle multiple objectives with the use of the utility-based compromise Decision Support Problem.

Following the presentation of the second example problem in Chapter 5, the theoretical performance validity is examined in the final chapter. At this point the thesis is critically evaluated. The relevant contributions are summarized and suggestions for future research will also be provided.

CHAPTER V

CUSTOMIZED PROCESS PLATFORM DESIGN: THE RAPID MANUFACTURING OF HEARING AID SHELLS

The augmented Product Platform Constructal Theory Method is presented in Chapter 3. In Chapter 4, the augmented PPCTM is applied to the realm of product platform design for a line of customizable pressure vessel. The use of the PPCTM to develop a product platform in a market of non-uniform demand is validated through the pressure vessel example problem. In this chapter, the augmented PPCTM is applied to the design of a manufacturing process for customizable products. Specifically, the augmented PPCTM's ability to handle multiple objectives and the realm of process design will be tested through the development of a manufacturing process family for the production of customizable hearing aid shells.

The primary focus in this chapter is to provide empirical support for Hypotheses 2 and 3 (from Section 1.3):

Q2: *How can the Product Platform Constructal Theory Method be augmented in order to enable a designer to cope with the presence of multiple objectives?*

H2: The infusion of the utility-based compromise Decision Support Problem will provide a rigorous construct in which a designer can make decisions with multiple, conflicted, and coupled objectives.

Q3: *How can the Product Platform Constructal Theory Method be successfully applied to the domain of production process development for customized products in a market of non-uniform demand?*

H3: The generalized application of the PPCTM (and the augmentations proposed in H1 and H2) can be applied to assist a designer with the development of a "process platform" for the manufacture of customized products.

The secondary focus in this chapter is the continued establishment of empirical structural and empirical theoretical validity of the augmented PPCTM (Figure 5.1). By applying the augmented PPCTM to this example problem, confidence in the appropriateness of the example and confidence in the usefulness of the method will be established.

The customized hearing aid shell example problem is introduced in Section 5.1. The description of the application of the augmented PPCTM to this example problem is detailed in Section 5.2. The description of the specific solution method and the presentation of results are then presented in Section 5.3. Finally, in Section 5.4, a summary of the chapter and a look ahead to the closing chapter is provided.

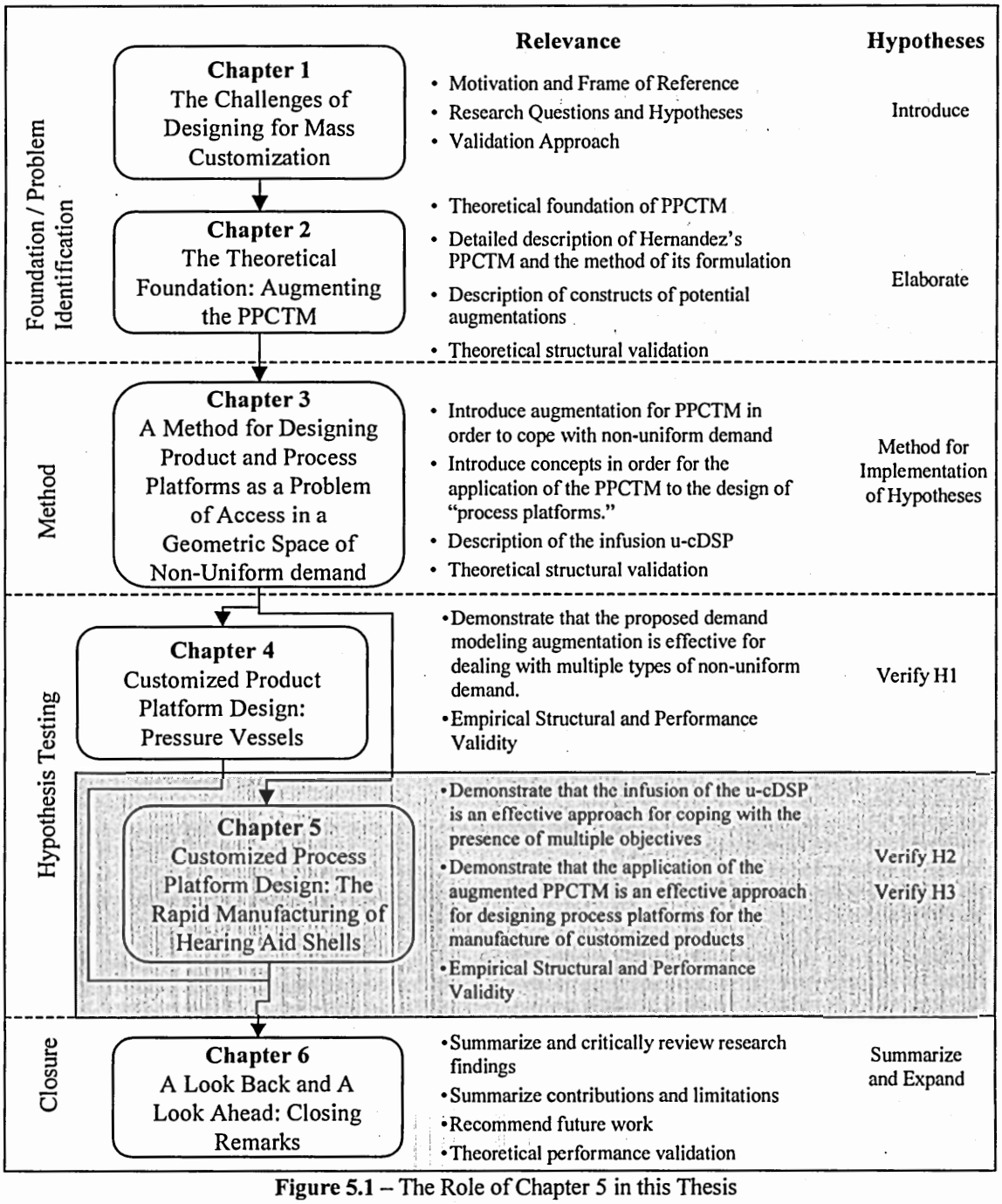


Figure 5.1 – The Role of Chapter 5 in this Thesis

5.1 THE PROBLEM OF PRODUCING CUSTOMIZED HEARING AID SHELLS IN A MARKET OF NON-UNIFORM DEMAND

Consider a manufacturer of hearing aids. This manufacturer seeks a competitive advantage over other leading manufacturing enterprises through offering personalized hearing aids. After significant research, the manufacturer discovers that through the use of a rapid prototyping technology, thousands of unique hearing aids can be efficiently and effectively produced. Specifically, the manufacturer establishes the following plan for customized hearing aid shell production:

- After consulting with a doctor, a clay impression of the patient's ear canal can be created.
- Using a 3D laser scanning technology, the impression can be digitized and easily converted into a CAD file
- The CAD file is modified to model a personalized hearing aid shell for the patient.
- Each patient's CAD file is be manufactured using Stratasys' Fused Deposition Modeling (FDM) (Stratasys, 2003) technology (discussed in detail in Section 5.1.2).

The manufacturer believes that the use of the FDM rapid prototyping technology will provide the manufacturing enterprise the agility and flexibility to offer a personalized hearing aid to each customer at a competitive price. When compared to traditional hard tooling alternatives, rapid prototyping (in general) offers a number of advantages in the quest to create customizable geometry affordably:

- Rapid prototyping gives the manufacturer the ability to create multiple, different geometries in a single build. A traditional casting process would require the creation of a separate mold for each desired geometry. The RP engineer is capable of producing many different parts in one build by simply adding each of their 3D CAD models to the build file.
- Rapid prototyping does not require a change in tooling for the creation of different geometries. Each technology uses a central method in which to "draw and fill" the cross section of each part. Unless the parts are of different materials, many parts can be manufactured in one common build.
- The desire to created customized geometries with rapid prototyping does not require extra skilled operator input. To manufacture different geometries with RP technology, the operator is simply required to orient the part in the build file.

This particular hearing aid example is based on an actual current product line produced by a collaboration between Simens and Phonak (Masters, 2002).

While the manner in which product variety will be offered has been determined, the specific manner in which the manufacturing process will be configured has not. The manufacturing engineer faces a dilemma, as the selection of process parameters for Fused Deposition Modeling is a very difficult and complex problem, especially when confronted with the need to configure the process to satisfy three conflicting objectives. Specifically, the manufacturing engineer must determine an appropriate batch size, layer thickness, road width, machine, and the appropriate number of machines for a certain level of product demand. These choices are directed towards three objectives: the minimization of production cost, the minimization of production time, and the

maximization of the quality of each part. The manufacturer is also faced with a serious production constraint: the parts must be completed no more than one week from the date they were ordered.

After consulting with the marketing department of this company, the manufacturer discovers that the existing problem of process design is even more complicated than imagined. The demand for this customized product is highly non-uniform. As a result, each time the production capacity changes, the manufacturer will have to re-select all of the process parameters.

The manufacturing engineer is in search of a rigorous method in which a production *process platform* can be generated for the range of capacity requirements presented by this problem. Specifically, the engineer wishes to commonalize certain process parameters across the range of capacity so that the production process will not have to be reconfigured for changes in demand. Thus, the core problem to be addressed in this example is the use of the augmented PPCTM to develop a process platform from which a series of capacity requirements are fulfilled through altering sets of process parameters. The goal in the development of this process platform is to provide the best compromise to three conflicting objectives: the minimization of average cost and production time, and the maximization of the average part quality.

There are two key assumptions in the formulation of this example problem:

1. Product variety only through geometry customization: This manufacturing enterprise seeks to offer variety only through hearing aid shell geometry. Offering variety through a change in product material, product color, or in functionality is not considered in this example problem.

2. Complete and certain information: All relevant information and models needed to apply the augmented PPCTM are available, complete, and certain. The role of uncertainty and risk is not taken into account in the formulation of the problem. Conceptually, the infusion of the u-cDSP into the PPCTM should provide a designer with the power to model uncertainty; however, uncertainty in both goals and parameters is beyond the scope of this work.

A detailed description of the hearing aid market is given in Section 5.1.1. The series of models needed to approximate the Fused Deposition Modeling process are presented in Section 5.1.2. An overview of the Fused Deposition Modeling technology is also presented in Section 5.1.2 as necessary background information. The establishment of this context will provide the foundation on which the application of the augmented PPCTM will be presented.

Most of the modeling techniques presented in this work are directly leveraged from the author's previous work (Williams et al., 2003) and are cited where appropriate.

5.1.1 Hearing Aid Shell Market Description

Through review of market research, the manufacturing engineer discovers that demand for customized hearing aid shells range from an arrival rate of 120 parts per day to 1000 parts per day. This highly non-uniform demand for this customized product translates directly into a wide variety of production requirements for the rapid manufacturing process.

It is assumed it is technically feasible to produce all amounts of demand within this range. It is also assumed that, for a specified arrival rate, demand is constant. The

forecasting of demand is not of such high fidelity as to model an arrival rate dependent on certain hours of the day.

5.1.2 Hearing Aid Shell Process Model

For this example, it is assumed that the design of each hearing aid shell is completed by the time it arrives in the production queue. Each hearing aid shell consists of a hollow, negative impression of the customer's ear canal, as can be seen in the illustration shown in Figure 5.2.

Figure 5.2 – Hearing Aid Shell (Masters, 2002)

The exterior geometry is unique to the customer, however the internal void in the part must be large enough to house a common set of electronic components. For this reason, the majority of the hearing aid shells are of the same overall dimensions. In this example problem, each hearing aid shell is approximated as a truncated elliptical cone. Measurements, taken directly from an actual shell with micro-calipers, are displayed in Table 5.1. This model provides insight into the selection of some process parameters. For example, the extremely fine wall thickness provides a constraint on the feasible road width values.

Table 5.1 – Hearing Aid Shell Average Dimensions (Williams et al., 2003)

Hearing Aid Shell Model	**Upper Ellipse Minor Diameter** 10.72 mm	**Upper Ellipse Major Diameter** 18.7 mm
	Lower Ellipse Minor Diameter 6.36 mm	**Lower Ellipse Major Diameter** 10.5 mm
	Part Height 21.78 mm	**Wall Thickness** 1.11 mm
	Volume of Material	229 mm^3

An Overview of Fused Deposition Modeling

The Fused Deposition Modeling (FDM) process, developed by Stratasys, produces prototype parts by extruding a heated filament through a nozzle that lays down the part's cross-section layer by layer. As illustrated in Figure 5.2, the material filament is fed through a heating element where it becomes semi-molten. The filament is then fed through a nozzle and deposited onto the partially constructed part. The newly deposited material fuses with the adjacent material that has already been deposited. The machine deposits a layer of the part by moving the nozzle head around in the X-Y plane. Multiple layers are fused together to complete the prototype by lowering the build platform along the Z-axis.

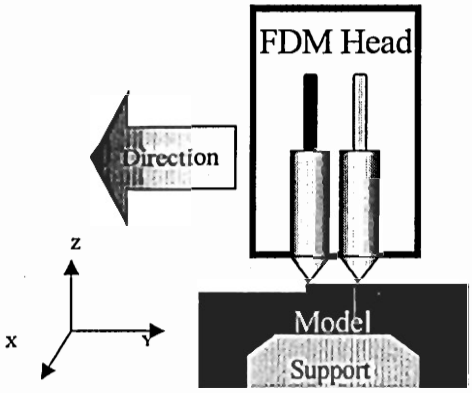


Figure 5.3 – The Fused Deposition Modeling Process (Williams et al., 2003)

As can be seen in Figure 5.3, FDM machines have a second nozzle that extrudes support material to enable the construction of overhanging structure.

For this example problem, three FDM machines are investigated for application: the Prodigy Plus, the FDM Titan, and the FDM Maxum. Each machine is modeled with specific process characteristics based on information received from the manufacturer. The maximum number of parts is estimated with the knowledge of the average hearing aid dimensions (discussed below), and the dimension of the build volume of each machine. This data is presented in Table 5.2.

Table 5.2 – Machine Process Characteristics (Stratasys, 2003)

Machine	*Max. Parts*	*Scan Speed*	*Layer Thickness*	*Road Width*	*Cost*
Prodigy Plus	234	64 mm/s	0.178 – 0.33 mm	0.19 – 0.21 mm	$70,000
Titan	818	127 mm/s	0.24 – 0.26 mm	0.19 – 0.21 mm	$210,000
Maxum	1703	254 mm/s	0.127 – 0.25 mm	0.193 – 0.965 mm	$260,000

These machine characteristics provide the foundation for the process constraints that are imposed throughout the formulation of this problem.

As stated above, the manufacturing engineer wants to develop a process platform for a range of capacity requirements. Specifically, the engineer wishes to specify the following process parameters: batch size (number of parts per build), layer thickness (the height of each individual layer deposit by the process), and the road width (the width of material deposited by the nozzle). The selection of these parameters is directed towards the improvement of three objectives: minimization of production time, minimization of production cost, and the maximization of part quality. With an understanding of the working principle of Fused Deposition Modeling, the manner in which these objectives are modeled is presented.

Production Time Model

The total time to build a batch of hearing aid shells with FDM is approximated as:

$$t_{batch} = t_s + \left(t_l N_l + t_{base}\right) N_{pb} \qquad [5.1]$$

where:

t_s = Setup time that includes heating the filament, setting the foam, etc. ($\cong$ 0.5 hr)

t_l = Time to build each layer

N_l = Number of layers

t_{base} = Time to build the base of the part (generally, 4 layers of support material are built)

N_{pb} = Number of parts per batch

The number of layers (N_l) needed to complete a part is the quotient of the height (h) of the part and the layer thickness (t_{layer}).

$$N_l = \frac{h}{t_{layer}} \qquad [5.2]$$

The time to build each layer is approximated as:

$$t_l = t_{lower} + t_{boundary} + t_{fill} + t_{supp} \qquad [5.3]$$

where :

t_{lower} = Time to move down z-axis one layer thickness

$t_{boundary}$ = Time to draw outer and inner boundaries of layer

t_{fill} = Time to fill the inner portion by raster scan

t_{supp} = Time to draw the support material for this layer

t_{lower} represents the time for the deposition head to move down one layer. For most FDM machines, the nozzles are cleaned after the deposition of two layers; it is estimated that this cleaning process takes 10 seconds to complete. To calculate the amount of lowering time for one layer, this cleaning time is spread over each layer; thus t_{lower} is estimated to be 5 seconds.

The time required to draw the boundary of the average part is proportional to the length of the contour. Since the shape of a hearing aid model is represented as a cone, the

average circumference is approximated as the product of the average length and width of the model:

$$t_{boundary} = \frac{\left[\left((l_u + l_l)/2\right)\left((w_u + w_l)/2\right)\right]}{V_{ss}} \qquad [5.4]$$

where the u and l subscripts represent the upper and lower ellipse dimensions of the cone (from Table 5.1), and V_{ss} represents the scan speed of the machine (from Table 5.2).

The time to fill the boundaries of each layer is calculated as:

$$t_{fill} = \frac{\text{Cross Sectional Area}}{\text{Road With * Scan Velocity}} = \frac{(t_s - 2w_{road})(l + w)}{w_{road} V_{ss}} \qquad [5.5]$$

where t_s is the shell thickness of the hearing aid shell (Table 5.1), w_{road} is the road width for the fill, and l and w are the average length and width of the shell model.

T_{supp} is the time required to build the support structure for each layer. Support structure is needed because of the overhang present in the hearing aid model. Support structures must be able to be easily removed after the build is completed; as such, there is a large distance between adjacent layers of support material. For this example, the time to build the support material is estimated as 25% of the time required to fill each layer (Equation 5.6).

$$t_{\text{supp}} = 0.25 t_{fill} \qquad [5.6]$$

While it is naïve to assume that the time to build each layer will be the same for each layer of each hearing aid shell, it is understood that this will not significantly influence the overall results or interfere with the validation of the augmented PPCTM.

The final portion of Equation 5.1 yet to be explained is the time required to build the base of each part. Generally, four layers of support material are laid down to provide a solid build foundation for each part. The time required to build these supports is:

$$t_{\text{base}} = 4\left(\frac{\left(\frac{\text{Support Area}}{\text{Road Width}}\right)}{\text{Scan Velocity}}\right) = 4\left(\frac{\left(\frac{lw}{w_{road}}\right)}{V_{ss}}\right) \qquad [5.7]$$

Modeling the cost of this production process is heavily dependent on production time. With a manner in which to model production time established, the cost model is now presented.

Production Cost Model

The cost of building a single batch of parts is estimated as:

$$C_{batch} = \left(V_{material}C_{material}\right)N_{pb} + \left(C_{labor} + C_{operation}\right)t_{batch} + \left[\frac{\left(C_{maint} + C_{machine}\right)N_{machine}}{N_{ppy}}\right]N_{pb} \qquad [5.8]$$

where:

$V_{material}$ = volume of material used to build one part (~229 mm^3)

$C_{material}$ = cost of material (includes build and support material) (build: 4.31 \$/in^3, ABS; support: 3.79 \$/in^3, WaterWorks)

C_{labor} = cost of labor (\$25 / hour)

$C_{operation}$ = hourly cost of machine operation (\$80 / hour)

C_{maint} = annual machine maintenance cost (\$50,000 / year)

$C_{machine}$ = cost of purchasing machine, to be paid through one year of production (see Table 5.2)

N_{ppy} = number of products produced in one year (= D_d * 365 day/year)

For this example, it is assumed that all parts will be made with ABS plastic. The hourly and annual fees for labor, operation, and maintenance are estimates of actual costs. Their

values do not change the fundamental validation strategy of applying the augmented PPCTM to this example problem.

Quality Model

For many products, quality is a metric with a subjective nature. When manufacturing with rapid prototyping, quality can be quantified by the "stair stepping" effect – the inability to produce smoothed surfaces in the z-direction due to the layer-based additive manufacturing method. For the purpose of this example problem, the quality metric is dependent on the selection of the layer thickness (LT) and road width (RW) process parameters. The manner in which different road widths and layer thicknesses affect the "stair stepping" effect is shown in Figure 5.4.

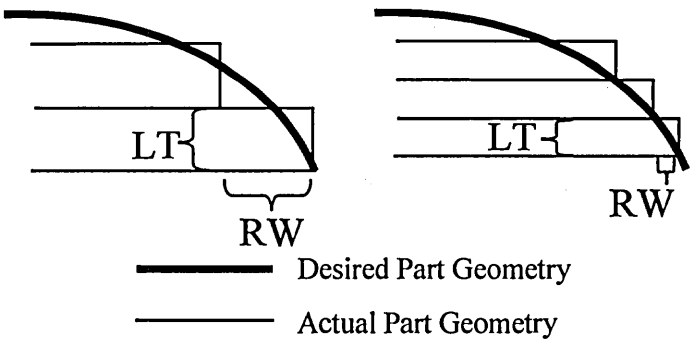


Figure 5.4 – Quality Metric Diagram

As can be seen, the best quality is achieved with a minimal layer thickness and minimal road width. Without an approximation of the curvature of the surface of each hearing aid shell, the quality metric can be simply modeled as the sum of these two process variables. The engineer would like to reduce this value of the quality metric for obtaining a good quality part. The quality metric, Q, is determined by the equation:

$$Q = t_{layer} + w_{road} \quad (mm) \qquad [5.9]$$

This method of quantifying quality is the most relevant for this application. Ideally one would compare the error between the actual part and the model via an exact representation of the curvature of the hearing aid.

Modeling Demand

As stated above, the capacity requirement for the production process of this example changes frequently. As such it is beneficial to model demand information in units of demand per day. This simulates an arrival rate of demand to the rapid prototyping machines.

Given the batch size, and the necessary capacity requirement, one can calculate the number of batches required to meet this demand in one week's time:

$$N_b = \frac{7D_d}{N_{pb}} \qquad [5.11]$$

In order to deal with the queuing and batching of the arriving parts, the concept of "wait time" is introduced. The wait time, t_w, is the manner in which the machine's creation of batches are modeled. t_w is the amount of time the manufacturer waits before beginning the production of a single batch. As time passes, more and more parts arrive at the workstation (as a function of demand per day, D_d):

$$t_w = \frac{N_{pb}}{D_d}(24*3600) \ \text{(s)} \qquad [5.10]$$

Through the substitution of these equations into the cost and time equations previously developed, demand and the number of batches are related to time and cost.

Establishing Constraints

In order to meet the requirements of the problem statement, the manufacturer imposes four constraints to be certain that:

(1) the parts are produced within a week of their order arrival

(2) the batch size is not greater than the maximum allowed in the machine

(3) the cost of the process is not too expensive

(4) the parts are of acceptable quality

(1) To assure that the parts are produced within a week of their order arrival, the total cycle time of the production of a product is calculated as the sum of the total time required to build a capacity requirement and the time required to wait for a batch to be built:

$$t_{cycletime} = t_w + (N_b t_{batch}) \qquad [5.11]$$

This value of cycle time must be less than one week for the solution to be considered feasible. Any combination of process parameters that provide a total cycle time more than the lead time of one week are immediately discarded. Following the ideas of mass production, it is obvious that larger batch sizes translate into lower costs (the machine cost is spread out over a larger number of parts). Without this constraint, the solution algorithm would select the largest batch size allowed by each machine without concern for the amount of time needed to queue such a large batch.

(2) As presented in Table 5.2, each machine has an estimate of the maximum number of parts per batch that it can physically build. This constraint provides an upper bound on the selection of the batch size for the process.

(3) & (4) The use of utility metrics is one manner in which constraints are applied to the solution space. Any combination of build parameters that provides a negative utility value is immediately ignored by setting the expected utility value to zero. This applies bounds on the cost and quality per part – any combination of parameters that is

too expensive or produces low-quality parts cannot be accepted. A description of these individual utility functions are presented in Section 5.2.4.

The six-step method described earlier in Section 3.4 is applied to this example problem in Section 5.2. The focus in this example is the application of the augmented PPCTM in the realm of process design to facilitate the development of a process platform for a line of customizable hearing aid shells in a scenario of non-uniform capacity requirements. The resultant process platform will consist of a set of baseline process parameters from which multiple levels of demand can be satisfied.

5.2 EXPRESSING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE

In this section the manner in which the augmented Product Platform Constructal Theory Method (PPCTM, presented in Chapter 3) is applied to the design of a process platform for a line of customizable hearing aid shells is presented. The six steps of the augmented PPCTM are presented in Figure 5.5.

In this section, the application of the first five steps is presented. The sixth step, the solution of the multistage utility-based compromise Decision Support Problem, is discussed in detail in Section 5.3. The presentation of this chapter is similar to that of Chapter 4; however, the differences between the design of the product and the process platform are highlighted.

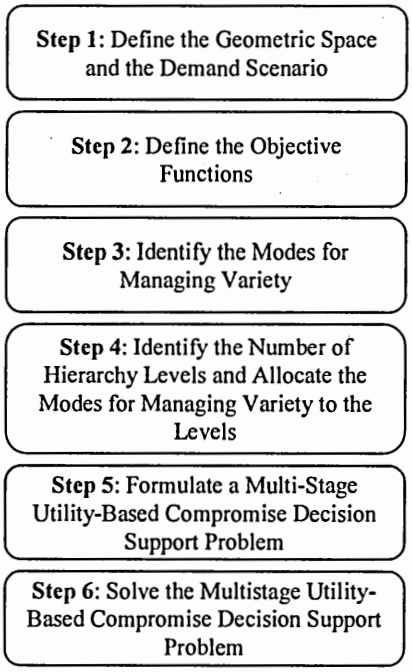


Figure 5.5 – The Six Steps of the Augmented PPCTM

5.2.1 Step 1: Define the Geometric Space and the Demand Scenario

The first step of the augmented PPCTM is the abstraction of a geometric space from the problem. For product design this space is defined as the space of customization and is defined by the design specifications for which variety will be offered. Applying the augmented PPCTM to process design, a designer must define a similar geometric space. When applied to the development of a process platform, this geometric space is called a *space of capacity*. As defined in Section 3.2.2, a space of capacity is the range of capacity that a manufacturing enterprise is willing to satisfy with a process platform. This space is typically a one-dimensional space that is bounded by the manufacturer's selection of a range of capacity to be offered by the particular production process.

Through collaboration with the marketing team, the manufacturer has expressed a requirement that the production process of customized hearing aid shells must be able to handle a demand from 120 parts per day to 1000 parts per day. The resulting space of capacity is provided in Figure 5.6.

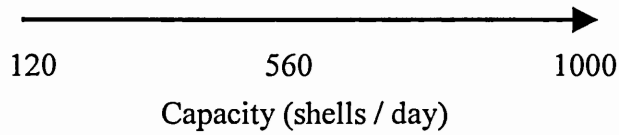


Figure 5.6 – Hearing Aid Shell Space of Capacity

Unlike the application of the augmented PPCTM to product design, the notion of non-uniform demand is inherently captured with the application of the method to process platform development. Each point along the space of capacity represents a different level of demand. In fact, the motivation for the development of a process platform is centered on the presence of non-uniform demand. In the use of the PPCTM for product platform design, the platform was developed for a "snapshot" of demand. For process design, there is no need to forecast or model demand; instead the platform is developed in order to deal with a changing demand over a period of time.

5.2.2 Step 2: Define the Objective Functions

As stated in the problem description in Section 5.1, the manufacturing engineer wishes to find the best compromise between three conflicting objectives: to minimize the cost of the production process, to minimize the amount of time to build a batch of parts, and to maximize the quality of each part. For the development of a process platform, a engineer wishes to optimize the average value of these objectives across each capacity requirement.

This example problem is formulated in a similar fashion as the pressure vessel example problem of Chapter 4. As presented throughout the thesis, the decision for each stage is dependent on the analysis of the values of the objective functions for the entire space, and can be calculated using either an integral or a discretized approach. The complexity of the analysis of the objective function calculations of this specific example problem suggests that a discretized approach is more useful. By discretizing the space of

customization and analyzing the objective functions at each "node," the overall utility of the process family can be calculated easily by averaging the objectives across the space and then calculating the resulting expected utility.

The focus in this particular step of the augmented PPCTM is to define the necessary objective functions. The calculation of average time for the entire process family is based on the amount of time to build one batch, t_{batch}, of a particular capacity requirement from Equation 5.1. The average build time of the process family is calculated as:

$$t_{avg} = \frac{\left(\sum_{i=D_{\min}}^{D_{\max}} t_{batch,i} N_{batch,i}\right) + \sum_{j=1}^{N_{setup}} t_{setup,j}}{D_{\max} - D_{\min}} \qquad [5.12]$$

where D_{max} and D_{min} represent the upper and lower bounds of the capacity space respectively. This build time metric also includes a time penalty, t_{setup}, of 30 min. This penalty is accrued for each different arrangement of process parameters, N_{setup}, across the process family.

The calculation of the average cost of the process family is performed in a similar fashion. From the cost per batch of a single capacity requirement, C_{batch} (Equation 5.8), the average cost is calculated as:

$$C_{avg} = \frac{\left(\sum_{i=D_{\min}}^{D_{\max}} C_{batch,i} N_{batch,i}\right) + \left(\sum_{j=1}^{N_{setup}} C_{setup,j}\right)}{D_{\max} - D_{\min}} \qquad [5.13]$$

A penalty for the setup of a different arrangement of process parameters is also added to this metric with C_{setup}, which is equal to \$50.

Finally, the average quality of the process family is calculated using Equation 5.9, the quality of a single part:

$$Q_{avg} = \frac{\sum_{i=D_{\min}}^{D_{\max}} Q_i}{D_{\max} - D_{\min}} \qquad [5.14]$$

Unfortunately, these objectives are inherently contradictory. One cannot maximize one, without minimizing another (and vice versa). For example, to maximize the quality, one would set the layer thickness and road width parameters to a minimum. This however, would drastically increase material costs as well as production time. In order to find the best compromise for these tradeoffs, an engineer must define an expected utility function. This formulation is done in Step 5 of this method.

5.2.3 Step 3: Identify the Modes for Managing Variety

For product platform design, the focus in the use of the PPCTM is the organization and the synthesis of modes for managing product variety – approaches in product design for achieving customization systematically. In Section 3.2.3, the concept of *modes for managing process variety* is defined to apply the PPCTM to process platform design. A mode for managing process variety is any generic approach in the design of a production process for achieving a change in manufacturing capacity.

For this third step in the augmented PPCTM, the designer identifies those modes that will be used to offer variety in the production capacity of the manufacturing process. Since the space for capacity is one-dimensional, each mode will be directed towards achieving variety in capacity only. For this example problem, five different modes for managing process customization are implemented.

<u>Mode D1: Customization of the Batch Size</u>

One mode of managing variety in capacity for this example problem is the standardization of the batch size. With the use of this mode, large changes in capacity are

achieved by choosing a different batch size. The determination of batch size is a very important parameter decision. As discussed above, larger batch sizes are preferred because of the associated decrease in costs. The batch size cannot be too large as it takes time to wait to queue a large batch of parts; one must be careful not to overstep the weeklong production time constraint.

Mode D2: Standardization of the Layer Thickness

Another mode for managing variety of capacity is to standardize the layer thickness of each batch. For this mode, changes in capacity are achieved by altering this common layer thickness while maintaining a constant road width and batch size. The selection of layer thickness is very important to the overall process; its direct correlation to the number of layers to be built by the process affects both the production time and cost estimates (Equations 5.1 and 5.8, respectively). High fidelity changes in capacity can be achieved by slightly tuning this process parameter. Of course, layer thickness plays an important role in the quality metric as well (Equation 5.9). As such, the selection of an appropriate layer thickness must be based on the outcome of all three objectives.

Mode D3: Standardization of the Road Width

The third mode for managing variety in capacity for this example problem is through the standardization of the road width across the range of capacity. Similar to layer thickness, road width directly affects the outcome of all three objective functions. Unlike layer thickness, changes in the road width parameter do not translate into large changes in the ability of a process to handle a different capacity. Different capacities are achieved with this mode by commonalizing the road width across a given batch size.

Mode D4: Commonalization of Machine Type

Changes in capacity are also achieved by changing the type of machine used in the production process. Changing the machine type changes the maximum batch size constraint, and also affects the speed at which the layers of the parts can be drawn. Of course, different machines cost different amounts of money, so the objective of minimizing cost is severely altered with each different machine type. Similar to all forms of technology, larger, faster machines cost more money and typically offer better production quality. As such, the selection of machine type is crucial. The use of this mode for managing variety commonalizes the scan speed and maximum parts allowed per batch over a range of capacity needs.

Mode D5: Altering the Number of Machines

This fifth and final mode of managing capacity for this example problem is the selection of the number of machines used for the production process. This mode is similar to Mode V2 of the pressure vessel example (Section 4.2.3); it involves the notion of modular combination to achieve variety. For this example, "modular combination" refers to the addition of machines to compensate for large demands of capacity. Since it is assumed that only similar machines can be added to the production process, the addition of a machine simply doubles the capacity of the manufacturing process. This doubling comes at a large cost however.

These five modes of managing process customization are the approaches considered for accessing all the points of the space of capacity generated in Step 1 of this example. Next, in Steps 4 through 6, a hierarchic organization of these modes is synthesized in order to offer a variety of production capacity in the space of customization.

5.2.4 Step 4: Identify the Number of Hierarchy Levels and Allocate the Modes for Managing Variety to the Levels

The synthesis of the multiple modes for managing variety presented in the previous section begins with determining the hierarchical order of each mode.

The First Stage and the First Space Element

As seen in the application of the augmented PPCTM to the pressure vessel example problem (Section 4.2.4), the mode of managing customization that offers the smallest variations in the dimension(s) of the space of customization is to be used first in the hierarchy. For this example problem, of all the process parameters to be varied, altering the batch size provides the best "control" over changes in production capacity. Customizing the batch size provides the simplest (changing the batch size does not require additional operator input) and most cost effective manner (there are no setups costs associated with changing the batch size) for a designer to offer a specific production capacity requirement.

The mode D1, "Customization of the Batch Size," is used to define a batch size for a capacity D_2 that, with a constant layer thickness and road width, will satisfy any capacity $D_1 \leq D_2$. The size of this first space element is given by the value of the variable ΔD_1, as shown in Figure 5.7.

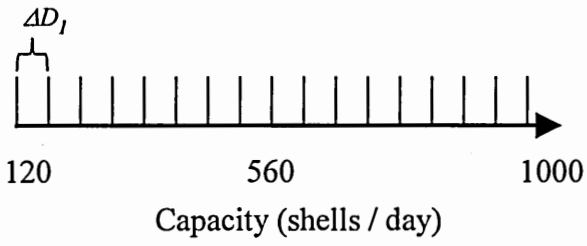


Figure 5.7 – The First Space Element of the Hearing Aid Space of Capacity

The Second Stage and the Second Space Element

The mode D2, "Standardization of Layer Thickness," is chosen as the mode for managing variety for the second stage in the hierarchy. Of the modes remaining for selection, this altering the layer thickness of the part provides a designer with the ability to make smaller adjustments in the level of production capacity. From Equations 5.1 and 5.2 one observes that the value of layer thickness chosen has a dramatic effect on the production time of a part. Distinct changes in production capacity are achieved by carefully selecting the layer thickness. A larger layer thickness shortens the build time and provides a decrease in cost; however, the average part quality will significantly decrease (Equation 5.9).

Mode D2 is used to define a layer thickness for a capacity D_2 that, with a constant batch size and road width, will satisfy any capacity $D_1 \leq D_2$. The size of this second space element is given by the value of the variable ΔD_2, as shown in Figure 5.8.

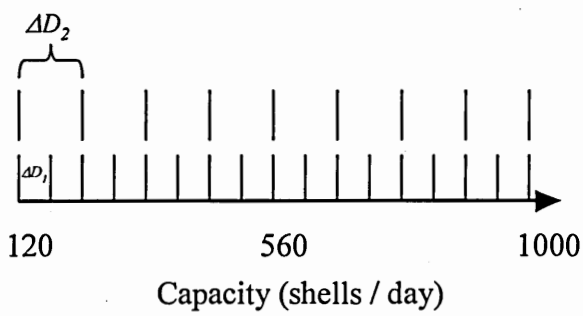


Figure 5.8 – The Second Space Element of the Hearing Aid Space of Capacity

Each second space element is composed of a number of first space elements.

The Third Stage and the Third Space Element

This third stage follows the same formulation as found in the previous stages. The mode D3, "Standardization of Road Width," is used to define a road width for a capacity D_2 that, with a constant batch size and layer thickness, will satisfy any capacity $D_1 \leq D_2$.

Changing the value of road width does not provide as such high fidelity changes in production capacity as the previous two modes do. Since the value selected for the road width parameter does not have as big of an impact as layer thickness, a designer cannot achieve a large variety of values of production capacity. As such, mode D3 is placed at a higher level of the hierarchy than mode D2.

Furthermore, similar to layer thickness, increasing the road width of a part decreases the build time and cost, as well as lowers the average part quality. These three metrics are not as sensitive to changes in road width as they are with changes in layer thickness.

The size of this third space element is given by the value of the variable ΔD_3, as shown in Figure 5.9.

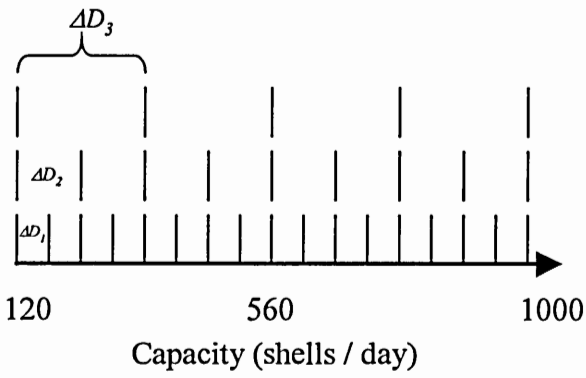


Figure 5.9 – The Third Space Element of the Hearing Aid Space of Capacity

The Fourth Stage and the Fourth Space Element

The fourth and final space element, S_4, is composed of a number of assemblies of the third space element, S_3, as shown in Figure 5.10.

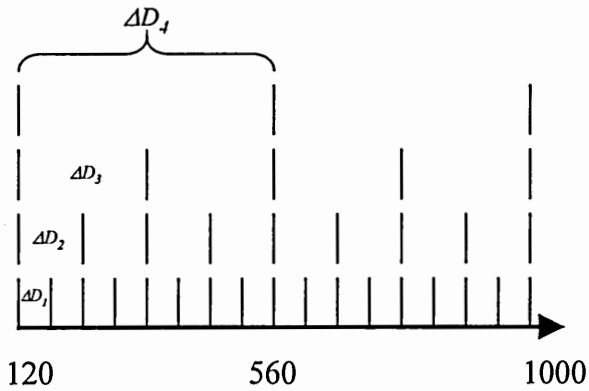


Figure 5.10 – The Fourth Space Element of the Hearing Aid Example

In this final space the remaining two modes of managing process customization are used. Mode D4, "Commonalization of Machine Type," is utilized in this space element along with Mode D5, "Altering the Number of Machines."

The application of this mode is different than the application of the other modes in the previous stages. In fact, the implementation of this mode is similar to the third stage of the pressure vessel example (see Section 4.2.4). Further correlations between these two stages can be made by drawing parallels between the concept of modularly combining components to offer larger values of the design parameter (i.e., volume in Chapter 4's pressure vessel, and for this example problem, production capacity).

Mode D4 is based on the idea that certain capacity requirements are suited for different types of machines. Whether limited by maximum build size or by a slow scan speed, lower-end machines (i.e., the Prodigy Plus and FDM Titan) simply cannot produce parts at a sufficient rate to meet larger demands. This applies to the other end of the capacity space spectrum as well; higher-end machines (i.e., the FDM Maxum) may be too expensive to justify their use for lower capacity needs. Mode D5, the combination of similar machines, is investigated in this space as an opportunity to offer more capacity. Increasing the number of machines produces an interesting tradeoff: it not only increases

the amount of capacity of the production process by decreasing the build time, but it also increases the costs associated with purchasing and maintaining the machines. Since these two modes are only able to offer expensive, large, discrete changes in production capacity, they are used at the highest level of the hierarchy.

Obviously, a machine setup that can satisfy a capacity D_2 will also satisfy any capacity D_1, where $D_1 \leq D_2$. Therefore, the focus in this decision stage is the assignment of different machine types and quantities to specific ranges of the capacity space. This is achieved by identifying six cutoff points along the space. As can be seen in Figure 5.11, each cutoff point represents a different combination of machine type and quantity.

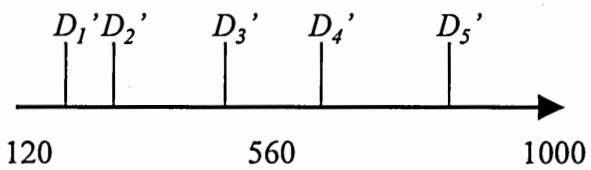


Figure 5.11 – Visualization of the Fourth Stage of the Hearing Aid Example Problem

where $D \leq D_1'$ One Prodigy Plus

$D_1' < D \leq D_2'$ Two Prodigy Plus

$D_2' < D \leq D_3'$ One FDM Titan

$D_3' < D \leq D_4'$ Two FDM Titans

$D_4' < D \leq D_5'$ One FDM Maxum

$D_5' < D$ Two FDM Maxums

This specific ordering of each cutoff point is based on the maximum number of parts per build of each machine type/quantity combination (see Table 5.2).

The selection of the type and number of machines for the manufacturing process is very important. The machines' scan speed and vat size (which relates to maximum

number of parts per build) directly correlates to the ability of each machine to produce the required number of parts within a week. Each individual machine has a different range of feasible road widths and layer thicknesses from which a manufacturing engineer can choose. Finally, each machine has a specific cost that plays a very important role in the final calculation of average cost of the process. The machines investigated for this example range in price from $70,000 to $260,000. This final decision stage provides a foundation on which the remaining of the decisions is based.

With all of the modes for managing variety successfully organized, the formulation of the multi-stage u-cDSP begins. A graphical interpretation of the modes for managing process variety is provided in Figure 5.12.

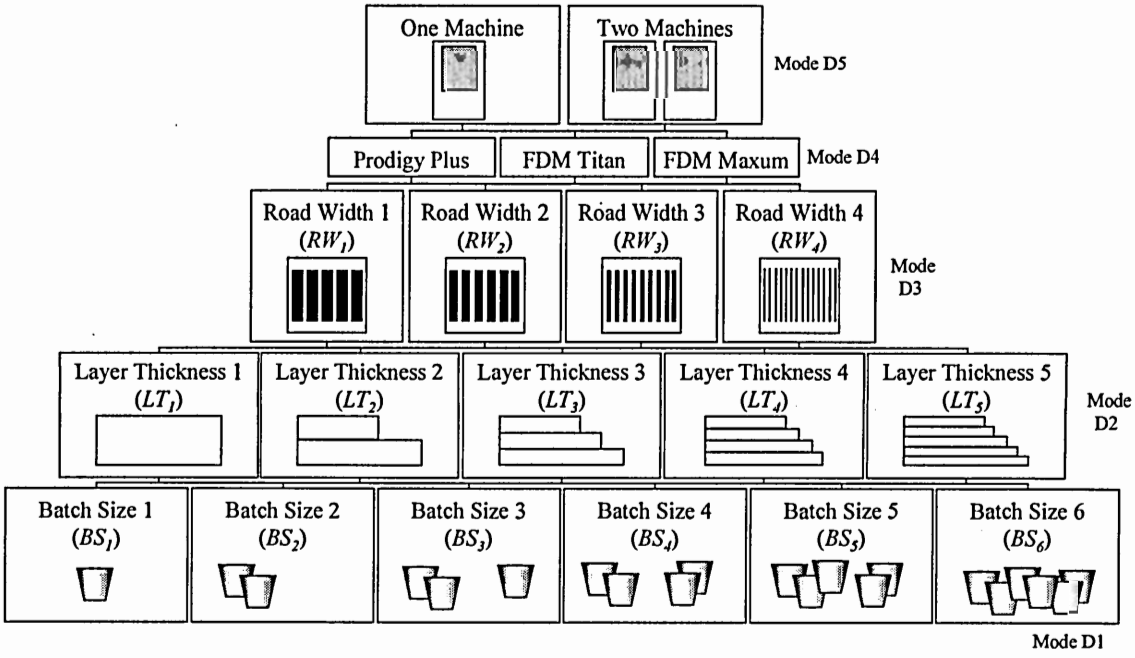


Figure 5.12 – Hierarchic Organization of the Modes for Managing Process Customization for the Hearing Aid Example

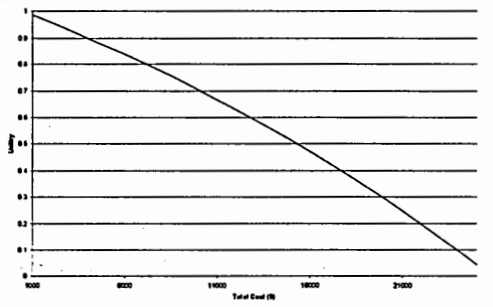
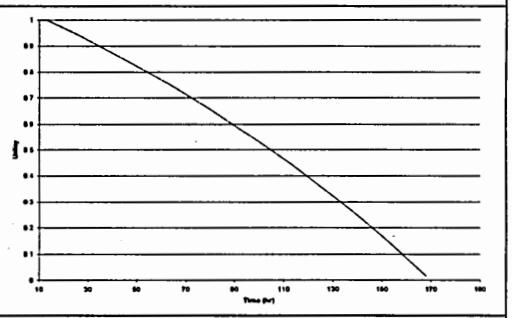
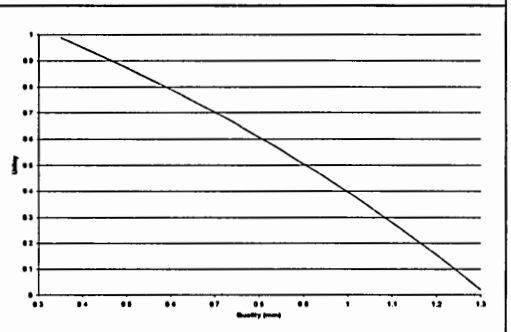
5.2.5 Step 5: Formulate a Multi-Stage Utility-Based Compromise Decision Support Problem

For this fifth step, a designer must formulate a multi-stage utility-based compromise Decision Support Problem based on the hierarchy developed in the previous step. The formulation of a u-cDSP is accomplished through following four steps (Section 3.3.1):

- assess utility functions for each goal,
- combine utility functions for individual goals into a multi-attribute utility function,
- formulate system goals, and
- formulate the deviation function.

The same multi-attribute utility function is formulated for each decision stage because their focus is in achieving the same three goals (i.e., the minimization of average cost, average time, and maximization of average part quality). The multi-attribute utility function for each stage is based on the individual utility functions of cost, time, and quality. The values of time, cost, and quality to be used in utility assessment are calculated using the average values calculated in each decision stage (presented later in this section). Each individual utility function is presented in Table 5.3.

Table 5.3 – Utility Functions for Cost, Time, and Quality

Utility Value	*Parameter Value*	*Utility Function*	*Utility Plot*
Cost ($), $k_C = 0.25$			
1	500	$u(C) = 2 - e^{\left(\frac{0.685(C - C_{\min})}{(C_{\max} - C_{\min})}\right)}$	
0.75	8500		
0.5	15250		
0.25	21000		
0	25000		
Time (hrs), $k_t = 0.25$			
1	13	$u(t) = 2 - e^{\left(\frac{0.685(t - t_{\min})}{(t_{\max} - t_{\min})}\right)}$	
0.75	63		
0.5	105		
0.25	139		
0	168		
Quality (mm), $k_q = 0.5$			
1	0.35	$u(q) = 2.04 - e^{\left(\frac{0.6861(q - q_{\min})}{(q_{\max} - q_{\min})}\right)}$	
0.75	0.65		
0.5	0.9		
0.25	1.12		
0	1.3		

where the *min* and *max* subscripts refer to the values of each parameter with utilities of 0 and 1 respectively.

The *k*-values presented in the table above are calculated by solving a system of equations similar to that which is presented in the second step of the utility-based compromise Decision Support Problem in Section 2.2.3. The *k*-values are used in concert with the individual utility functions for the determination of the overall expected utility, $E[U(D_d)]$.

$$U(D_d) = k_C u(C) + k_t u(t) + k_q u(Q) \qquad [5.15]$$

The goal of each objective is to maximize the value of each individual utility function (i.e., for the value to reach the ideal value, 1).

$$E[u(C_i)] + d_{C_i}^- + d_{C_i}^+ = 1 \qquad [5.16]$$

$$E[u(t_i)] + d_{t_i}^- + d_{t_i}^+ = 1 \qquad [5.17]$$

$$E[u(q_i)] + d_{q_i}^- + d_{q_i}^+ = 1 \qquad [5.18]$$

The deviation function for the multi-attribute utility function of each decision stage, therefore, is to minimize the deviation of the expected utility function from the ideal value:

$$Z_i = 1 - E[U_i(D_d)] = \sum_j^3 k_j (d_j^- + d_j^+) \qquad [5.19]$$

With the general formulation of the u-cDSP presented, the details of each stage's specific formulation are presented below.

"Decision 0"

During this fifth step of the augmented PPCTM, a designer must formulate a decision for each stage wherein the range of each mode of managing customization is selected. Inherent in these decisions is the selection of the actual value of the variable to be commonalized across each sub-space. In the pressure vessel example of Chapter 4, the determination of the value of the product parameter to be commonalized was simply governed by design constraints. For this example, this convenience is not present. Due to the multiple objectives involved with this example, one cannot identify a specific value of the process parameter that satisfies production constraints and also minimizes cost and time, and maximizes quality without a detailed analysis.

The value of each commonalized process parameter is based on the largest capacity of the subspace. This concept is based on the fact that any combination of process parameters that can satisfy a demand D_2 can also satisfy any demand D_1 where $D_1 \leq D_2$. In order to determine the appropriate values of the parameters to be commonalized for each subspace, a designer must employ a utility-based compromise DSP. This initial decision is presented in Figure 5.13 as "Decision 0."

Given:	Capacity Requirement; D_d (parts /day) Machine Type; Prodigy Plus, FDM Titan, or FDM Maxum Machine Number; N_{mach}	
Find:	Batch Size; N_{pb} Layer Thickness; t_{layer} Road Width; w_{road} Deviation variables; d_i^- and d_i^+	
Satisfy:	Bounds:	$0 \leq N_{pb} \leq N_{pb,max}$ $t_{layer,min} \leq t_{layer} \leq t_{layer,max}$ $w_{road,min} \leq w_{road} \leq w_{road,max}$
	Constraints:	$t_{cycletime} \leq 1$ week $d_i^-, d_i^+ \geq 0$ $d_i^- d_i^+ = 0$
	Goals:	$E[u(C)]+d_C^- - d_C^+ = 1$ $E[u(t)]+d_t^- - d_t^+ = 1$ $E[u(Q)]+d_Q^- - d_Q^+ = 1$

Minimize: $Z = 1 - E[U(D)] = 1 - \left[k_C(d_C^- - d_C^+) + k_t(d_t^- - d_t^+) + k_q(d_q^- - d_q^+)\right]$

Figure 5.13 – Decision Formulation for "Decision 0" of Hearing Aid Example

The *max* and *min* subscripts refer to the imposed bounds of the machine for each process parameters (see Table 5.2). The individual utility functions of Decision 0 (E[u(C)], E[u(t)], E[u(q)]) are evaluated using Equations 5.1, 5.8, and 5.9 respectively.

Decision 0 will be used inherently in each of the decision stages in order to determine the value of the parameter that will be commonalized across each subspace. Further discussion of the use of this decision stage is provided in Section 5.2.6.

The First Stage and the First Space Element

The first space element is defined by the range of application of Mode D1: customization of the batch size. ΔD_1 determines the extent of the commonality of the batch size, and is therefore the decision variable for this first stage. It is recalled from the second step, Section 5.2.2, that the focus in this decision is the determination of ΔD_1 that provides the best value for the overall objective – the maximization of the expected utility of all the objectives.

In this first stage, the calculation of cost, time, and quality at each node begins with the determination of the value of the batch size that should be commonalized. As stated above, this value is determined at the maximum capacity value of the current sub-space. Given the layer thickness and road width, the batch size that provides the largest expected utility within the space is solved for using the u-cDSP outline in "Decision 0" (Figure 5.13).

This common batch size is then applied to each capacity requirement of the first space. The resulting time, cost, and quality metric values are then averaged across the first space. This is done by first evaluating the number of first space elements in the space of capacity:

$$N_1 = \frac{(D_{\max} - D_{\min})}{(D_{\max,1} - D_{\min,1})} = \frac{(1000 - 120)\ parts/day}{(D_{\max,1} - D_{\min,1})} \qquad [5.20]$$

The values of $D_{max,1}$ and $D_{min,1}$ are based on the corresponding value of $\Delta C1$:

$$D_{\max,1} = D_{\min,1} + \Delta D_1 \qquad [5.21]$$

The average time of one first space element is:

$$t_{avg,1} = \frac{\left(\sum_{i=D_{\min,1}}^{D_{\max,1}} t_{batch,i} N_{batch,i} \right)}{D_{\max,1} - D_{\min,1}} \qquad [5.22]$$

Similarly, the average cost of a first space element is:

$$C_{avg,1} = \frac{\left(\sum_{i=D_{\min,1}}^{D_{\max,1}} C_{batch,i} N_{batch,i} \right)}{D_{\max,1} - D_{\min,1}} \qquad [5.23]$$

Finally, the average quality of a first space element is calculated by:

$$Q_{avg,1} = \frac{\sum_{i=D_{\min,1}}^{D_{\max,1}} Q_i}{D_{\max,1} - D_{\min,1}} \qquad [5.24]$$

The average time of all of the first space elements is calculated as:

$$t_{avg} = \left(\frac{1}{D_{\max} - D_{\min}} \right) \left(\sum_{i=1}^{N_1} t_{avg,1,i} + \sum_{j=1}^{N_{setup}} t_{setup,j} \right) \qquad [5.25]$$

Similarly the average cost of the first space elements is calculated with:

$$C_{avg} = \left(\frac{1}{D_{\max} - D_{\min}} \right) \left[\left(\sum_{i=1}^{N_1} C_{avg,1,i} \right) + \left(\sum_{j=1}^{N_{setup}} C_{setup,j} \right) \right] \qquad [5.26]$$

The third objective, the maximization of average quality, is averaged across the space of capacity via:

$$Q_{avg} = \left(\frac{1}{D_{\max} - D_{\min}} \right) \left(\sum_{i=1}^{N_1} Q_{avg,1,i} \right) \qquad [5.27]$$

These averaged values are then used to calculate an expected utility of the first space. The individual utility functions (*u(C)*, *u(t)*, and *u(Q)*), and the resulting expected utility function (*E[U(D)]*), are calculated as presented in Table 5.3. The resulting decision formulation for the first stage is shown in Figure 5.14.

Given:	The one-dimensional capacity space Mode D1: Customization of the Batch Size	
Find:	The value of the decision variable ΔD_1 Deviation variables, d_C^-, d_C^+, d_t^-, d_t^+, d_Q^-, d_Q^+	
Satisfy:	Bounds:	$0 \leq \Delta D_1 \leq 880$
	Constraints:	$t_{cycletime} \leq 1$ week $d_i^-, d_i^+ \geq 0$ $d_i^- \square d_i^+ = 0$
	Goals:	$E[u(C)]+d_C^- - d_C^+ = 1$ $E[u(t)]+d_t^- d_t^+ = 1$ $E[u(Q)]+d_Q^- d_Q^+ = 1$
Minimize: $Z_1 = 1 - E[U(D_d)] = 1 - \left[k_C(d_C^- - d_C^+) + k_t(d_t^- - d_t^+) + k_q(d_q^- - d_q^+)\right]$		

Figure 5.14 – Decision Formulation for the First Space Element

Inherent in this first stage decision is the value of layer thickness and road width process parameters, as well as the type and number of machines used in the production process. The individual utility functions cannot be evaluated without these values. These crucial details are determined in the following decisions.

<u>The Second Stage and Second Space Element</u>

The range of application of Mode D2, standardization of the layer thickness, defines the size of the second space element. Each second space element is composed of a number of first space elements. The number of first space elements that compose a second space element, $N_{1,2}$, is calculated as:

$$N_{1,2} = \frac{S_2}{S_1} = \frac{\Delta D_2}{\Delta D_1} = \frac{D_{\max,2} - D_{\min,2}}{D_{\max,1} - D_{\min,1}} \qquad [5.28]$$

The range of commonality of the layer thickness process parameter is defined by the decision variable ΔD_2. Similar to the previous decision stage, "Decision 0" is used to identify the layer thickness that provides the best compromise between the three conflicting objectives for the largest capacity requirement of each second sub-space. This value is then commonalized over a series of capacity requirements as dictated by the value of ΔD_2.

Similar to the first stage, the focus of this second decision stage is the selection of a value of ΔD_2 that maximizes the expected utility of the space of customization defined in Step 1. The calculation of the average time, cost, and quality of each second space element is dependent on the number of first space elements that compose the second space element:

$$t_{avg,2} = \frac{\sum_{i=1}^{N_{1,2}} t_{avg,1,i}}{N_{1,2}} \qquad [5.29]$$

$$C_{avg,2} = \frac{\sum_{i=1}^{N_{1,2}} C_{avg,1,i}}{N_{1,2}} \qquad [5.30]$$

$$Q_{avg,2} = \frac{\sum_{i=1}^{N_{1,2}} Q_{avg,1,i}}{N_{1,2}} \qquad [5.31]$$

The average time of all of the second space elements is calculated with:

$$t_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left(\sum_{i=1}^{N_2} t_{avg,2,i} + \sum_{j=1}^{N_{setup}} t_{setup,j}\right) \qquad [5.32]$$

Similarly the average cost of the second space elements is calculated with:

$$C_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left[\left(\sum_{i=1}^{N_2} C_{avg,2,i}\right) + \left(\sum_{j=1}^{N_{setup}} C_{setup,j}\right)\right] \qquad [5.33]$$

The third objective, the maximization of average quality, is averaged across the space of capacity via:

$$Q_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left(\sum_{i=1}^{N_2} Q_{avg,2,i}\right) \qquad [5.34]$$

where N_2 is the number of second space elements within the entire capacity space.

$$N_2 = \frac{(D_{\max} - D_{\min})}{(D_{\max,2} - D_{\min,2})} \qquad [5.35]$$

Following the format of Stage 1, these averaged values are then used to calculate the individual utility functions (Table 5.3). Finally, these individual utility functions are combined into an overall expected utility function. The focus in the decision of this second stage (Figure 5.15) is the minimization of the deviation of this expected utility from 1.

Inherent in this second stage decision is the value of the road width process parameter, as well as the type and number of machines used in the production process. These crucial details are determined in the following decisions.

Given:	The one-dimensional capacity space Mode D2: Standardization of Layer Thickness The value of ΔD_1	
Find:	The value of the decision variable ΔD_2 Deviation variables, d_C^-, d_C^+, d_t^-, d_t^+, d_Q^-, d_Q^+	
Satisfy:	Bounds:	$0 \leq \Delta D_2 \leq 880$
	Constraints:	$\Delta D_1 \leq \Delta D_2 \leq 880$ $t_{cycletime} \leq 1$ week $d_i^-, d_i^+ \geq 0$ $d_i^- \square d_i^+ = 0$
	Goals:	$E[u(C)]+d_C^- - d_C^+ = 1$ $E[u(t)]+d_t^- d_t^+ = 1$ $E[u(Q)]+d_Q^- d_Q^+ = 1$

Minimize: $Z_2 = 1 - E[U(D)] = 1 - \left[k_C(d_C^- - d_C^+) + k_t(d_t^- - d_t^+) + k_q(d_q^- - d_q^+)\right]$

Figure 5.15 – Decision Formulation for the Second Space Element

The Third Stage and Third Space Element

For the third space Mode D3, standardization of road width, is used to offer variety in the production capacity of the manufacturing process. The range of commonality of road width, ΔD_3, determines the size of each third space element. Each third space element is composed of a number of second space elements. The number of second space elements that compose a third space element, $N_{2,3}$, is calculated as:

$$N_{2,3} = \frac{S_3}{S_2} = \frac{\Delta D_3}{\Delta D_2} = \frac{D_{\max,3} - D_{\min,3}}{D_{\max,2} - D_{\min,2}} \qquad [5.36]$$

The formulation of the decision third for the third stage is identical to that of the previous two. There are two decision variables for this third stage, the value of road width to be commonalized, and its range of commonality, ΔD_3. "Decision 0" is used to identify the road width that provides the best compromise between the three conflicting

objectives for the largest capacity requirement of each second sub-space. This value is then commonalized over a series of capacity requirements as dictated by the value of ΔD_3.

Similar to the first and second stages, the focus in this third decision stage is the selection of a value of ΔD_3 that maximizes the expected utility of the space of customization defined in Step 1 (Section 5.2.1).

The average time, cost, and quality of a third space element is shown in Equations 5.37 – 5.39.

$$t_{avg,3} = \frac{\sum_{i=1}^{N_{2,3}} t_{avg,2,i}}{N_{2,3}} \qquad [5.37]$$

$$C_{avg,3} = \frac{\sum_{i=1}^{N_{2,3}} C_{avg,2,i}}{N_{2,3}} \qquad [5.38]$$

$$Q_{avg,3} = \frac{\sum_{i=1}^{N_{2,3}} Q_{avg,2,i}}{N_{2,3}} \qquad [5.39]$$

The average time, cost, and quality of all of the third space elements is calculated with Equations 5.40 – 5.42.

$$t_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left(\sum_{i=1}^{N_3} t_{avg,3,i} + \sum_{j=1}^{N_{setup}} t_{setup,j}\right) \qquad [5.40]$$

$$C_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left[\left(\sum_{i=1}^{N_3} C_{avg,3,i}\right) + \left(\sum_{j=1}^{N_{setup}} C_{setup,j}\right)\right] \qquad [5.41]$$

$$Q_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left(\sum_{i=1}^{N_3} Q_{avg,3,i}\right) \qquad [5.42]$$

where N_3 is the number of third space elements within the entire capacity space.

$$N_3 = \frac{(D_{\max} - D_{\min})}{(D_{\max,3} - D_{\min,3})} \qquad [5.43]$$

Following the format of the previous stages, these averaged values are then used to calculate the individual utility functions presented in Table 5.3. Finally, these individual utility functions are combined into an overall expected utility function (Equation 5.15). The focus in the decision of this third stage (Figure 5.16) is the minimization of the deviation of this expected utility from 1.

Given:	The one-dimensional capacity space Mode C3: Standardization of Road Width The value of ΔD_1 The value of ΔD_2	
Find:	The value of the decision variable ΔD_3 Deviation variables, d_C^-, d_C^+, d_t^-, d_t^+, d_Q^-, d_Q^+	
Satisfy:	Bounds:	$0 \le \Delta D_3 \le 880$
	Constraints:	$\Delta D_2 \le \Delta D_3 \le 880$ $t_{cycletime} \le 1$ week $d_i^-, d_i^+ \ge 0$ $d_i^- \square d_i^+ = 0$
	Goals:	$E[u(C)]+d_C^- - d_C^+ = 1$ $E[u(t)]+d_t^- d_t^+ = 1$ $E[u(Q)]+d_Q^- d_Q^+ = 1$

Minimize: $Z_3 = 1 - E[U(D)] = 1 - \left[k_C(d_C^- - d_C^+) + k_t(d_t^- - d_t^+) + k_q(d_q^- - d_q^+)\right]$

Figure 5.16 – Decision Formulation for the Third Space Element

The Fourth Stage and Fourth Space Element

The size of the fourth and final space element is determined by the range of commonality of each machine type / quantity combination. The range of application of Modes D4 and D5 are defined by the placement of the "cut-off" points – the capacity

requirements that require a different machine type and/or quantity. As with the previous decision stages, the focus of the implementation of Modes D4 and D5 in this fourth stage is the maximization of expected utility for the entire capacity space. Similar to the previous steps, average time, cost, and quality are all calculated as a sum of the spaces of the previous stage. The analysis begins with the calculation of the number of third space elements in each fourth stage cutoff point, *4,i*.

$$N_{3,4_i} = \frac{S_{4,i}}{S_3} = \frac{\Delta D_{4,i}}{\Delta D_3} = \frac{D_{\max,4,i} - D_{\min,4,i}}{D_{\max,3} - D_{\min,3}} \qquad [5.44]$$

The average time, cost, and quality of a fourth space element is evaluated using Equations 5.45 – 5.47.

$$t_{avg,4_i} = \frac{\sum_{j=1}^{N_{3,4_i}} t_{avg,3,j}}{N_{3,4_i}} \qquad [5.45]$$

$$C_{avg,4_i} = \frac{\sum_{j=1}^{N_{3,4_i}} C_{avg,3,j}}{N_{3,4_i}} \qquad [5.46]$$

$$Q_{avg,4_i} = \frac{\sum_{j=1}^{N_{3,4_i}} Q_{avg,3,j}}{N_{3,4_i}} \qquad [5.47]$$

The average time, cost, and quality of all of the fourth space elements is calculated with Equations 5.48 – 5.50.

$$t_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left(\sum_{j=1}^{N_{4,i}} t_{avg,4_i,j}\right) \qquad [5.48]$$

$$C_{avg} = \left(\frac{1}{D_{\max} - D_{\min}}\right)\left[\left(\sum_{j=1}^{N_{4,i}} C_{avg,4_i,j}\right)\right] \qquad [5.49]$$

$$Q_{avg} = \left(\frac{1}{D_{\max} - D_{\min}} \right) \left(\sum_{j=1}^{N_{4_i}} Q_{avg,4_i,j} \right) \qquad [5.50]$$

where $N_{4,i}$ is the number of fourth space elements within the entire capacity space.

Following the format of the previous stages, these averaged values are then used to calculate the individual utility functions as shown in Table 5.3. Finally, these individual utility functions are combined into an overall expected utility function (Equation 5.15). The focus in the decision of this fourth stage (Figure 5.18) is the minimization of the deviation of this expected utility from 1.

Given:	The one-dimensional capacity space Mode C4: Commonalization of Machine Type Mode C5: Altering the Number of Machines The value of ΔD_1 The value of ΔD_2 The value of ΔD_3	
Find:	The value of the decision variable ΔD_4 The location of each cutoff point, D_1', D_2', D_3', D_4', D_5' Deviation variables, d_C^-, d_C^+, d_t^-, d_t^+, d_Q^-, d_Q^+	
Satisfy:	Bounds:	$0 \leq \Delta D_4 \leq 880$
	Constraints:	$\Delta D_3 \leq \Delta D_4 \leq 880$ $t_{cycletime} \leq 1$ week $d_i^-, d_i^+ \geq 0$ $d_i^- \square d_i^+ = 0$
	Goals:	$E[u(C)]+d_C^- - d_C^+ = 1$ $E[u(t)]+d_t^- d_t^+ = 1$ $E[u(Q)]+d_Q^- d_Q^+ = 1$
Minimize: $Z_4 = 1 - E[U(D)] = 1 - \left[k_C(d_C^- - d_C^+) + k_t(d_t^- - d_t^+) + k_q(d_q^- - d_q^+) \right]$		

Figure 5.17 – Decision Formulation for the Fourth Space Element

As there are no other modes of managing process customization remaining, this fourth stage is the last stage in the hierarchy. A summary of the decision variables for

each stage is presented in Table 5.4. The relationship between each stage is provided in Figure 5.19.

Table 5.4 – Decision Variables of Each Stage of the Hearing Aid Analysis

Stage	Mode of Managing Customization	Decision Variable	Objective
1	Mode D1	ΔD_1	Minimize deviation of expected utility function, Z_1, from 1 (Figure 5.14)
2	Mode D2	ΔD_2	Minimize deviation of expected utility function, Z_2, from 1 (Figure 5.15)
3	Mode D3	ΔD_3	Minimize deviation of expected utility function, Z_3, from 1 (Figure 5.16)
4	Modes D4	ΔD_4	Minimize deviation of expected utility function, Z_4, from 1 (Figure 5.17)

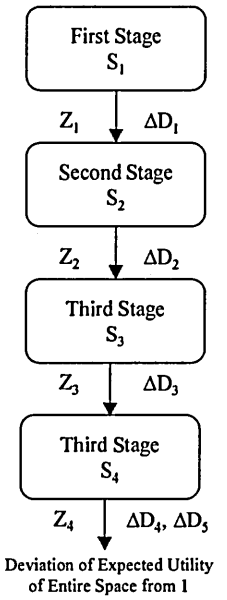


Figure 5.18 – Sequential Solution Process of the Four Stages of the Hearing Aid Example

From the review of constructal theory literature (Section 2.1) and from discussion in the previous example problem (Section 4.2.4), it is agreed that such a sequential solution process as shown in the above figure is not a sufficient means of solving this complex problem. An appropriate method for solving each of these decision stages in an uncoupled and recursive manner is provided in the Section 5.2.6.

The result of the solution process is the determination of the ranges of the modes of managing process customization that will produce the best compromise between the three objectives (minimize cost, minimize build time, and maximize quality).

5.2.6 Step 6: Solve the Multi-Stage Utility-Based Compromise Decision Support Problem

The final step of this augmented PPCTM is the formulation of an appropriate solution algorithm. Any appropriate solution technique can be used; the primary goal in the solution is the determination of the ranges of the modes that provide the largest expected utility (i.e., the best compromise between the conflicting objectives of minimizing cost, minimizing build time, and maximizing part quality). A graphical representation of the solution method for this problem is presented in Figure 5.19 (following the general solution method presented in Figure 3.14).

As described in Section 3.4.6, this solution method involves iterating through values of the modes of managing variety (ΔD_1, ΔD_2, ΔD_3, and ΔD_4), establishing the dimensions of the sub-spaces, commonalizing the design parameters (N_{pb}, t_{layer}, and w_{road}) across each sub-space, evaluating the objective functions, and comparing the resulting overall utility of each iteration. A suitable solution technique is employed in order to reevaluate the input based on its resultant deviation from the ideal expected utility value (see Section 5.3.2). Since no design constraints offer guidance for the determination of the values of each parameter to commonalize, an internal u-cDSP (dubbed "Decision 0", Section 5.2.5) is solved to inform the designer of the most appropriate values of each parameter to commonalize for each space. Similar to the multi-stage u-cDSP formulated for the solution of the overall problem, this "Decision 0" can be solved using any appropriate solution algorithm.

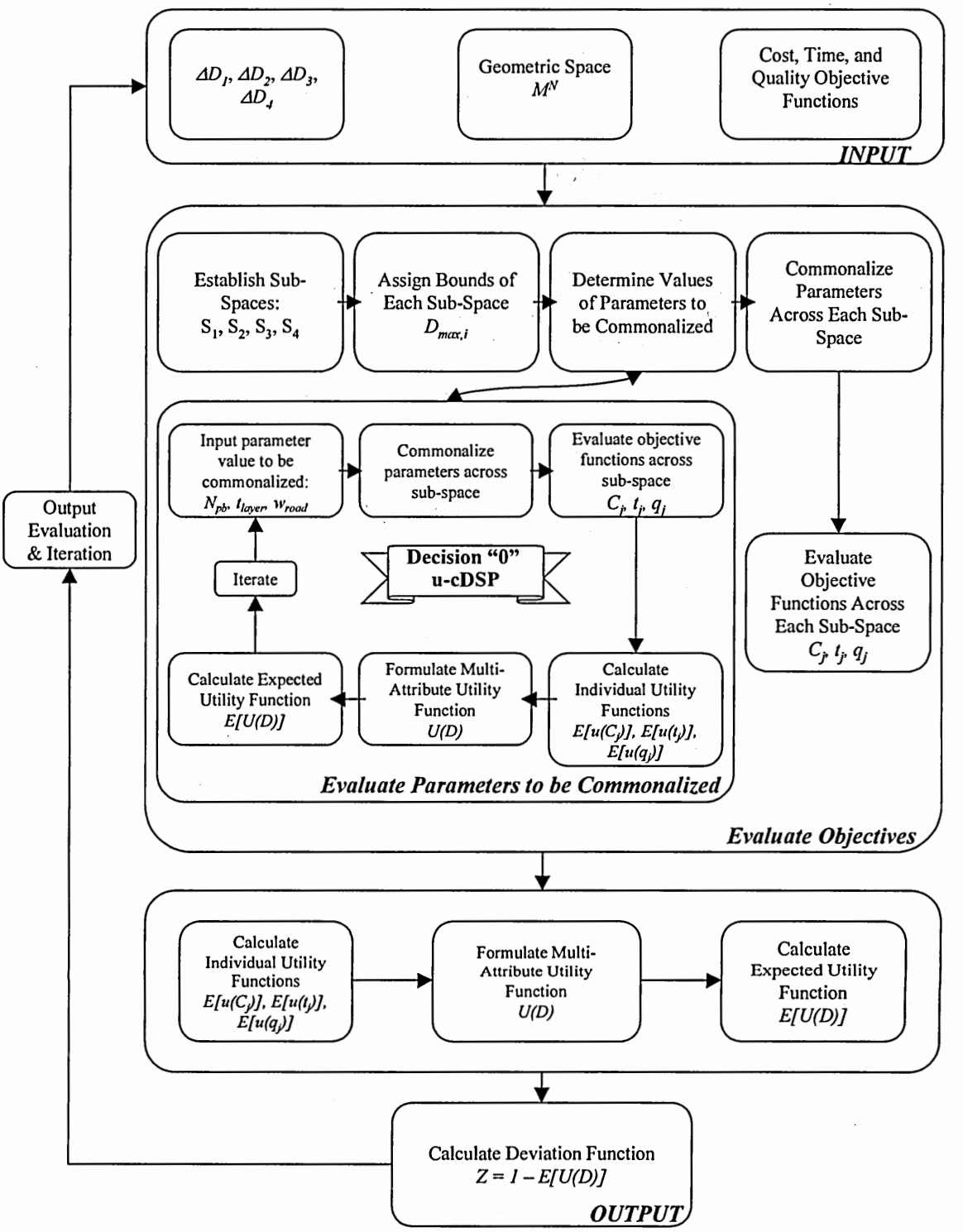


Figure 5.19 –Solution Algorithm for the Hearing Aid Example

From a higher level of abstraction, there are three major decisions in the solution algorithm of the hearing aid example, illustrated in Figure 5.20.

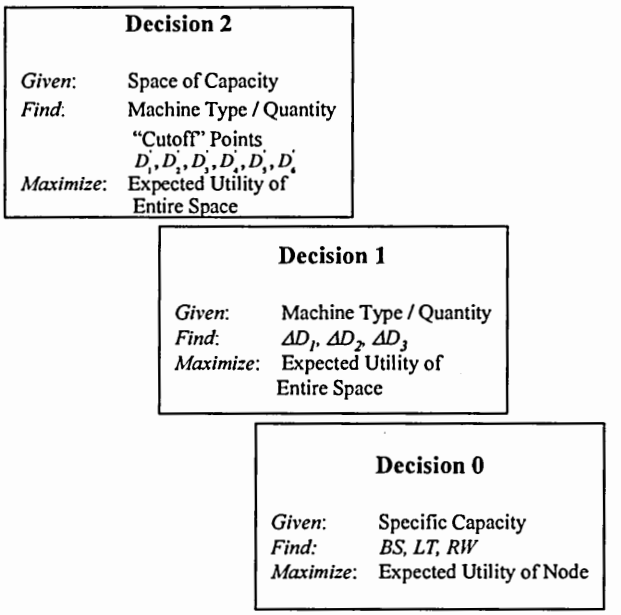


Figure 5.20 – The Three Major Decisions of the Hearing Aid Example

The coupling of the need to search for the values of process parameters to be commonalized at each subspace (Decision 0), with the need to investigate several different combinations of ranges of commonality for each mode (Decision 1) and for each machine type/quantity (Decision 2) makes this overall problem very complex. To provide further insight into this solution algorithm, the following tutorial example is offered.

Consider the initial guess for the range of applications for each mode for managing capacity variety shown in Table 5.5.

Table 5.5 – Example Values of Modes of Managing Process Variety

ΔD_1	4 parts / day
ΔD_2	10 parts / day
ΔD_3	20 parts / day

From these values, each sub-space can be determined, as shown in Figure 5.21 (a smaller space of capacity is considered for this tutorial example).

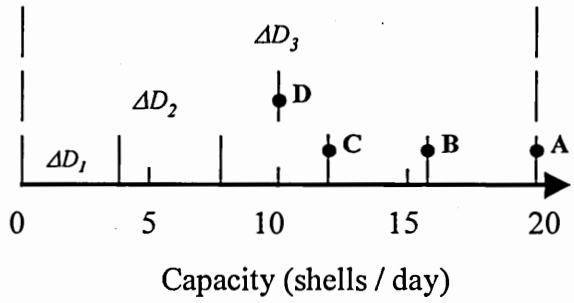


Figure 5.21– Example Space of Capacity for Illustration of Decision 0

Inherent in the determination of the sub-spaces is the identification of the bounds of each sub-space (i.e., the maximum capacity for each first sub-space is 4, 8, 12, 16, and 20 parts/day).

With the identification of the bounds of each sub-space, "Decision 0" is invoked in order to determine the process parameters to be commonalized for each sub-space. This internal u-cDSP is solved at each sub-space's bound. For example, consider point A in Figure 5.21: a "Decision 0" u-cDSP (Figure 5.19) is solved in order to find the appropriate batch size, layer thickness, and road width to commonalize across their respective sub-spaces. The next "Decision 0" u-cDSP will be solved at point B in Figure 5.21. This time, however, the layer thickness and road width are given (they are the same from the u-cDSP solved at point A). Point C in Figure 5.21 has a similar u-cDSP to solve. Point D, however, requires the solution of a u-cDSP where the road width (determined in u-cDSP at point A) and the batch size (determined in u-cDSP at point C) are known, and the appropriate layer thickness is determined. This pattern continues until the appropriate parameters to be commonalized for each subspace are determined.

Once this is completed, the average time, cost, and quality of the variants across the space of capacity are evaluated. These objective functions are then used to formulate the multi-stage u-cDSP of the problem as a whole. An expected utility is calculated, and a

new guess for the ranges of each mode for managing variety is input; thus completing the iterative nature of the solution algorithm.

This trend continues for each machine type/quantity combination (completing Decision 1, as presented in Figure 5.20). The expected utilities of each variant in the space of capacity for each machine type/quantity combination are then compared. This comparison provides a designer with the ability to determine the appropriate cut-off points for each machine combination. For example, assume that the following values for expected utility are obtained after completing "Decision 1" for the tutorial example presented above (Figure 5.21).

Table 5.6 – Example Results for Illustration of Decision 2

Capacity (prd/week)	*E[U(D)] One Prodigy Plus*	*E[U(D)] Two Prodigy Plus*	*E[U(D)] One FDM Titan*	*E[U(D)] Two FDM Titans*	*E[U(D)] One FDM Maxum*	*E[U(D)] Two FDM Maxums*
5	0.83	0.7	0.79	0.4	0.67	0.25
10	0.6	0.5	0.75	0.5	0.7	0.3
15	0.45	0.3	0.73	0.52	0.75	0.5
20	0.4	0.2	0.71	0.62	0.8	0.56

By comparing these results, a designer is able to assign the appropriate range of application of each machine combination. From these specific results, a designer would choose to use one FDM Maxum for a capacity $10 < D \leq 20$, one FDM Titan for a capacity $5 \leq D \leq 10$, and one Prodigy Plus for a capacity $0 < D \leq 5$.

Obviously, the use of a solution technique that is capable of minimizing the number of iterations in this complex solution algorithm is recommended. The specifics of the solution technique employed in this work are explained in detail in Section 5.3.

5.3 SOLVING THE EXAMPLE AS A PROBLEM OF OPTIMAL ACCESS IN A GEOMETRIC SPACE

The final step of the augmented PPCTM, as described in Section 3.4.6, is the solution of the multi-stage utility-based compromise Decision Support Problem. A general solution method for this example is presented in Section 5.2.6 (Figure 5.19). Since this example is much more complex than the pressure vessel example (Chapter 4), a gradient-based solution algorithm is performed (Section 5.3.2). In this section, details of the solution technique are presented. This is followed by a presentation and analysis of the results (Sections 5.3.3 and 5.3.4).

5.3.1 Discretization of the Geometric Space

There are two available approaches for analyzing a space of capacity: through continuous evaluation of the space, or through the discretization of the space (Section 3.4.2). While analyzing the space of customization as a continuous space presents the most mathematically rigorous and robust solution technique, it is also much more complex than analysis through discretization. For continuous analysis, each objective function must be expressed as a function of the design parameters that are varied (i.e., quality must be expressed as a function of capacity). This would require a significant effort in derivation and integration.

Lack of mathematical complexity is not the only reason for choosing a discretized solution approach for this example problem. The presence of discrete modes of managing process customization, namely Modes C4 and C5 (altering machine type and quantity), prevent a continuous approach. Each machine has significantly different properties and, as result, has a significant impact on the selection of appropriate process parameters.

The discretization of this example's space of capacity is similar to that performed for the pressure vessel example of Chapter 4. A designer must first choose an appropriate resolution for the spacing between each product variant. Nodes are then established across the space of capacity. These nodes provide distinct points from which the solution algorithm can use to calculate the three objective functions, their individual utilities, and the overall expected utility function.

Some limitations of this approach include:

- When first applying this analysis technique, a designer may not be confident in his/her selection of node resolution. Multiple solution iterations may be needed to find a resolution that is appropriate for the specific problem.
- The evaluation of all three objective functions at each discrete node is computationally inefficient when compared to the evaluation of one objective function that integrates across the entire space as performed with a continuous analysis.

It should be noted that the search for appropriate ranges of modes for managing variety is no longer constrained to being multiples of node resolution and factors of the space of capacity (as it was in the pressure vessel example of Chapter 4).

As explained in the pressure vessel example problem, the most apparent limitation of a discrete analysis is the number of calculations required to fully explore the design space (Section 4.3.1). It is the author's opinion, however, that this limitation is outweighed by the benefit of being able to analyze an example with discrete modes of managing process customization.

5.3.2 Applying a Gradient-Based Solution Algorithm

Due to the complexity of this problem, a gradient-based solution algorithm is employed in order to determine the best range of commonality for each mode of managing process customization. This solution algorithm provides the input variables shown in Figure 5.19 (namely, the range of application of each mode for managing process variety) based on the output given by the multistage decision process. The algorithm is therefore iterative; the multistage decision in this problem is solved by inputting a specific combination of ranges of modes of managing customization for each arrangement of machine type and quantity and calculating the expected utility of the entire space of capacity. The implementation of a gradient-based solution algorithm serves as a wrapper around the general solution algorithm; it also reduces the number of solution iterations. The iterative feedback loop of this solution algorithm is illustrated in Figure 5.22 (a view of the solution algorithm, Figure 5.19, from a more abstract level).

Fundamentally the gradient-based algorithm provides a set of input variables, runs through the multistage decision process (the three decisions of Figure 5.20, shown in the inner boxes of Figure 5.22), evaluates the output, and finally, provides a new set of input variables based on the output. Specific detail of the execution sequence is provided in Sections 5.2.6 and 3.4.6. As discussed in Section 5.3.4, various starting points for each set of input variables are tried in order to validate the results.

Similar to the previous example problem, this algorithm is implemented using the JAVA programming language. The accompanying JAVA code is presented in Appendix C.

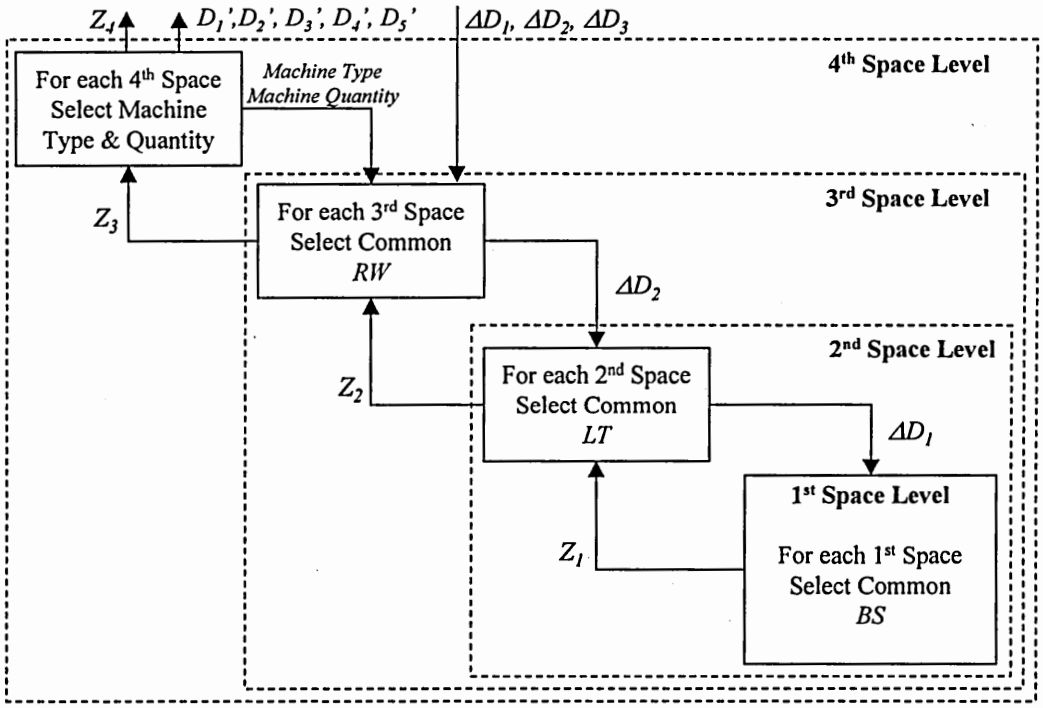


Figure 5.22 – Solution Process of Hearing Aid Example

Sequential Quadratic Programming Implementation

Instead of performing an exhaustive search of all possible combinations of ΔD_1, ΔD_2, and ΔD_3, a gradient-based solution algorithm is used to solve the multistage u-cDSP decision model. Specifically, Sequential Quadratic Programming (SQP) is used to find the best combination of ranges of each mode of managing process customization (Belegundu and Chandrupatla, 1999). The conditions of the SQP analysis are presented in Table 5.7.

Table 5.7 – Sequential Quadratic Programming Application Conditions

Maximum Number of Iterations	40
Termination Accuracy	0.000001
Relative Gradient Step	0.01
Minimum Absolute Gradient Step	0.001

For a successful application of SQP to this example problem, each decision variable must be normalized in order to ensure that the step size is appropriate and that the solution will converge appropriately.

The solution of this example problem using SQP is implemented using iSIGHT, a generic software shell that automates the iterative cycle of inputting variables, performing a simulation program, reading an output, adjusting the input, and so on. This cycle is illustrated in Figure 5.23.

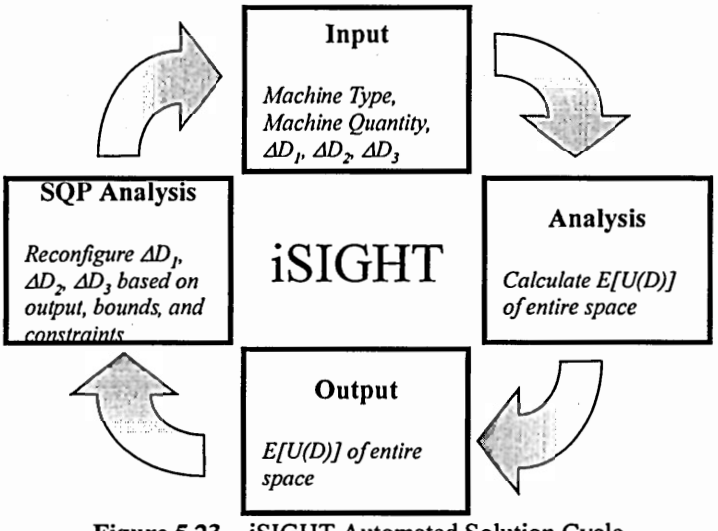


Figure 5.23 – iSIGHT Automated Solution Cycle

Dubbed as a "software robot" (Engineous, 2003), iSIGHT provides a foundation for the synthesis of simulation software and a variety of solution algorithms.

The verification and validation of this solution technique follows the presentation of the results in Section 5.3.4.

5.3.3 Evaluating the Results

As shown in Figure 5.20, the application of the augmented PPCTM to the hearing aid example is abstracted to three different major decisions:

- Decision 0: the selection of specific process parameters to be commonalized
- Decision 1: the selection of the range of commonalization for each mode of managing process customization
- Decision 2: the identification of appropriate "cutoff" points for each machine type / quantity combination.

In this section the results of each machine type and quantity combination are presented in order to verify and validate the method. The final decision, and the overall solution to this example, is then presented.

Decision 0 and Decision 1

The best combination of the first three modes of managing process customization for each machine is presented in Table 5.8. The average time (t_{avg}), average cost (C_{avg}), and average quality (Q_{avg}) of the entire space for each machine is also listed. The maximum capacity presented in the table represents the number of parts per day that each machine combination can feasibly build. The limitation exists for the smaller machines, as the machine's vat size is not large enough and the scan speed is not fast enough to build a larger quantity of parts within the week constraint. Finally, the overall expected utility of the entire space of capacity, and the deviation functions are presented.

Table 5.8 – Range of Commonality of the Modes of Managing Process Variety for Each Machine Combination of the Hearing Aid Example

Machine Type	Machine Quantity	ΔD_1 (ppd)	ΔD_2 (ppd)	ΔD_3 (ppd)	t_{avg} (hrs)	C_{avg} ($)	Q_{avg} (mm)	*Max. Capacity*	*E[U(D)]*	d^-	d^+
Prodigy Plus	1	5	44	44	100.65	13,252.83	0.5345	415	0.7325	0.2674	0
	2	5	125	125	92.18	24,667.95	0.5368	865	0.6085	0.3915	0
FDM Titan	1	5	93	80	88.82	14,856.97	0.4664	670	0.7623	0.2376	0
	2	5	440	440	62.59	23,888.68	0.4692	1000	0.6903	0.3097	0
FDM Maxum	1	5	21	21	47.42	11,659.32	0.5090	1000	0.8360	0.1640	0
	2	5	28	28	26.90	18,274.22	0.4760	1000	0.8042	0.1958	0

Observations of the results of this decision stage reveal a number of insights into the consistency of the formulation of this problem:

- Adding an additional machine to the process increases the maximum capacity of each machine and lowers the total time required to produce each level of demand. The cost, however, increases significantly due to the cost of an additional machine and its associated maintenance.

- The average time required to build a certain capacity is much larger for the Prodigy Plus than for the FDM Titan and Maxum. This is consistent with the data presented in Table 5.2 where it is observed that the lower-end machines have a much slower scan speed than that of the high-end machines. This slower scan speed increases the build time, and thus increases the operation and labor costs of the machine. As a result, the tradeoff between a machine's cost and its performance is not as significant as expected.
- The average quality of the higher-end machines is higher than the lower-end machines. The ability of the FDM Maxum to produce parts faster provides it the opportunity to make parts of higher quality.

Each of these observations is consistent with those predictions established in the modeling of the problem in Section 5.1.

In order for the ranges of each mode of managing process customization to be meaningful, they must be mapped onto the space of capacity (similar to what is done in Section 4.3.4 for the pressure vessel example). This mapping is provided for each machine in Appendix E. An example of mapping the chosen decision variables to the space of capacity is presented in Table 5.9 for one FDM Maxum.

Table 5.9 – A Sample of the Mapping Between the Process Platform and Parameters for One FDM Maxum

Capacity	Batch Size	Road Width (mm)	Layer Thickness (mm)	Cost ($)	Time (s)	Quality (mm)	E[U]
120	727	0.19	0.15	8765.42	91139	0.340	0.9550454
125	727			8765.42	91139	0.340	0.9550454
130	761			8874.12	94643	0.340	0.9529377
135	777			8984.78	98215	0.340	0.9507826
140	811			9143.48	103520	0.340	0.9476174
145	845		0.16	9033.32	99396	0.350	0.9463987
150	862			9138.04	102762	0.350	0.9443531
155	878			9242.71	106128	0.350	0.942301
160	912			9395.66	111234	0.350	0.9392265
165	946		0.17	9279.52	106897	0.359	0.938126
170	962			9378.99	110083	0.359	0.9361713
175	979			9478.41	113267	0.359	0.9342107
180	1013			9626.30	118198	0.359	0.9312167
185	1047		0.18	9506.08	113715	0.369	0.9301953
190	1063			9600.91	116740	0.369	0.928322
195	1080			9695.70	119764	0.369	0.9264432
200	1097			9840.46	124587	0.369	0.9234929
205	1164	0.22	0.17	9429.54	110302	0.395	0.9223631
210	1181			9513.25	112949	0.395	0.920723
215	1215			9595.69	115552	0.395	0.9191041
220	1232			9679.36	118197	0.395	0.9174555
225	1249			9813.00	122641	0.395	0.9147386
230	1282		0.18	9759.15	120455	0.400	0.9141156
235	1299			9840.89	123033	0.400	0.9124983
240	1316			9922.61	125611	0.400	0.9108768
245	1333			10054.31	129988	0.400	0.9081785
250	1383		0.19	9911.34	124708	0.410	0.907561
255	1400			9989.58	127165	0.410	0.9060092
260	1417			10067.81	129622	0.410	0.9044535
265	1434			10196.02	133878	0.410	0.9018165
270	1467		0.19	10137.38	131519	0.415	0.901292
275	1484			10213.98	133920	0.415	0.8997631
280	1518			10289.57	136285	0.415	0.8982517
285	1518			10417.15	140519	0.415	0.8956089
290	1568		0.20	10271.23	135125	0.425	0.8950539
295	1585			10344.87	137422	0.425	0.8935819
300	1602			10418.48	139719	0.425	0.8921064
305	1619			10542.09	143815	0.425	0.8895378

It is noted here that, in actual rapid prototyping process setup, an engineer does not have a choice of altering the layer thickness and road width to more than three significant digits. Also the engineer cannot establish a batch size that has a fraction of a part. These types of numbers are reported in the Table 5.9 in order to illustrate the ability of the method to select different parameters. If a designer wishes to select from a series of discrete values of process parameters, the augmented PPCTM, and the formulation of Decision 0, is able to handle that as well.

Decision 2

In order to complete the solution of this example problem, the cutoff points for each machine type and quantity are determined. This is accomplished through the comparison of the expected utility of each point in the capacity space for each machine. The machine type / quantity combination which provides the largest expected utility for a specific point in the capacity is then selected (Section 5.2.6). To illustrate this concept, another sample of a space of capacity roadmap, for one Prodigy Plus, is presented in Table 5.10.

Table 5.10 – A Sample of the Mapping Between Process Platform and Parameters for One Prodigy Plus

Capacity	Batch Size	Road Width (mm)	Layer Thickness (mm)	Cost ($)	Time (s)	Quality (mm)	E[U(D)]
120	234			7621.04	176596.9	0.504	0.8684775
125	234			7842.69	183955.1	0.504	0.8684775
130	234			8064.34	191313.3	0.504	0.8638643
135	234			8285.99	198671.5	0.504	0.8592145
140	234	0.21	0.29	8507.65	206029.7	0.504	0.8545277
145	234			8729.30	213387.9	0.504	0.8498038
150	234			8950.95	220746.1	0.504	0.8450424
155	234			9172.61	228104.3	0.504	0.8402431
160	234			9444.26	237262.5	0.504	0.834255

Through the comparison of the expected utility for each node of the space of capacity, a designer decides the appropriate machine type and quantity. Looking at 120 parts/day for

both the FDM Maxum (Table 5.9) and the Prodigy Plus (Table 5.10), it is observed that use of the FDM Maxum provides an increase in expected utility of 11% (0.955 vs. 0.869) due to its ability to offer a much lower average build time and higher average part quality. The comparison of each machine's expected utilities is provided in Appendix F.

Once the comparison of the expected utility of each node of the space of capacity for each machine is complete, the cutoff points for the implementation of each machine are established. The final result of applying the augmented PPCTM to the development of a process platform for the rapid manufacturing of customizable hearing aids is presented in Table 5.11.

Table 5.11 – Range of Each Mode of Managing Process Variety for Hearing Aid Example

Mode of Managing Customization	*Parameter Commonalized*	*Range of Commonalization (parts / day)*
ΔD_1	Batch Size	5
ΔD_2	Layer Thickness	21
ΔD_3	Road Width	21
D_1'	One Prodigy Plus	0
D_2'	Two Prodigy Plus	0
D_3'	One FDM Titan	0
D_4'	Two FDM Titans	0
D_5'	One FDM Maxum	880
t_{avg} = 47.42 hrs	C_{avg} = \$11,659.32	Q_{avg} = 0.509 mm
Expected Utility = 0.8360		

This table of results provides a manufacturing engineer with the range of commonalization for each mode of managing process customization. This is presented graphically in Figure 5.24.

To illustrate the significance of this result, the following tutorial example is presented. Imagine that the manufacturing enterprise reports an arrival rate of demand of 267 parts per day. Looking at Table 5.11, the manufacturing engineer approves the use of one FDM Maxum.

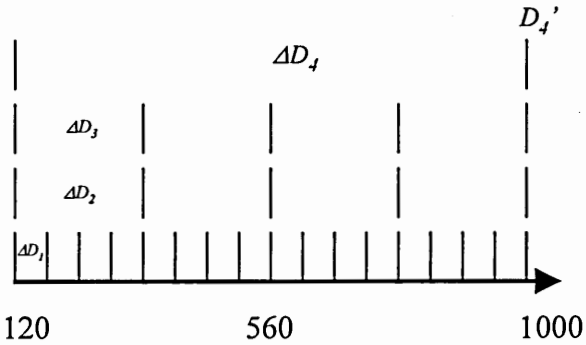


Figure 5.24 – Graphical Representation of Resultant Process Platform for Hearing Aid Example

Observing the results presented in Table 5.9 (or Appendix E.5), the manufacturer selects a batch size of 1468 parts/batch, a road width of 0.224 mm, and a layer thickness of 0.186 mm. The results presented in Table 5.9 also inform the manufacturer to expect a production time of 36.5 hrs and a cost of $10,137.38 in order to fulfill one week's capacity requirement. The quality of each part produced for this capacity will be 0.415 mm. If the manufacturing capacity requirement changes to 250 parts per day the following week, the engineer is only required to change the batch size (Table 5.9).

The validity of these results is analyzed in Section 5.3.4.

5.3.4 Verification of Results

For this example, the verification of the results is centered on the validity of using the Sequential Quadratic Programming solution algorithm (Section 5.3.2). The use of this solution technique is verified through the exploration of three different areas:

- using different starting points,
- using a different gradient-based solution algorithm, and
- verifying the results obtained with SQP using an exhaustive search.

<u>Altering the Starting Points</u>

In order to verify the results of each application of SQP, different starting points are input into the solution technique. The convergence values from each starting point are

then compared to each other in order to determine the best solution. Results are further verified by using converged values as new starting points. An example of this method of verification is provided in Table 5.12 below. Starting at an initial solution point of $\Delta D_1 = 5$, $\Delta D_2 = 50$, and $\Delta D_3 = 150$, the following results are provided by the application of SQP on one FDM Maxum.

Table 5.12 – Results of SQP from One FDM Maxum; Starting Point $\Delta D_1 = 5$, $\Delta D_2 = 50$, $\Delta D_3 = 150$

		INITIAL		*FINAL*
1	dBS	5		5
	dLT	50		21.487
	dRW	150		21.487
	E(U)			0.8360285
2	dBS	5		5
	dLT	21.487		20.99862
	dRW	21.487		20.99862
	E(U)	0.8360285		0.8360314
3	dBS	5		5
	dLT	20.99862		20.99862
	dRW	20.99862		20.99862
	E(U)	0.8360314		0.8360314

The first starting point converged to a solution of ΔD_2 and $\Delta D_3 = 21.5$ parts per day. Using this solution as a new starting point, the solution technique is run again, resulting in a new answer, $\Delta D_2 = \Delta D_3 = 20.99$ parts per day. A chart of the convergence of the deviation function for these series of starting points is provided in Figure 5.25.

Following this convergence, a completely new starting point (preferably distant from the previous in the solution space) is chosen. This type of verification is provided for each application of SQP on each machine type / quantity combination. The results and history of this verification is provided in Appendix D.

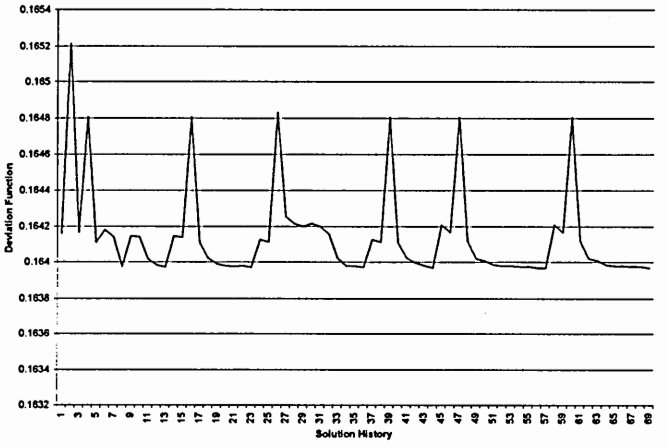


Figure 5.25 – Convergence History of SQP for One FDM Maxum

Sequential Quadratic Programming vs. Generalized Reduced Gradient Method

In order to verify that SQP is an appropriate gradient-based solution algorithm for this example problem, it is compared against results obtained with the Generalized Reduced Gradient (GRG) method (Belegundu and Chandrupatla, 1999).

Provided with identical starting points, the GRG fails to converge to a better solution than what was previously presented with the SQP. This can be observed by comparing Table 5.11 and Figure 5.25 with Table 5.12 and Figure 5.26.

Table 5.13 – Results of GRG from One FDM Maxum; Starting Point $\Delta D_1 = 5$, $\Delta D_2 = 50$, $\Delta D_3 = 150$

		INITIAL		*FINAL*
1	dBS	5		5
	dLT	50		50
	dRW	150		150
	E(U)			0.835839
2	dBS	5		5
	dLT	50		50
	dRW	150		150
	E(U)	0.835839		0.835839

As can be seen, the use of SQP provides a better solution than the use of GRG (expected utility of 0.8360 vs. 0.8358). SQP has a superior rate of convergence and typically requires fewer major iterations or trial solutions than a GRG method, thus

shortening iteration time (Belegundu and Chandrupatla, 1999). For these reasons, Sequential Quadratic Programming is used in the solution of the multi-stage utility-based Compromise Decision Support Problem.

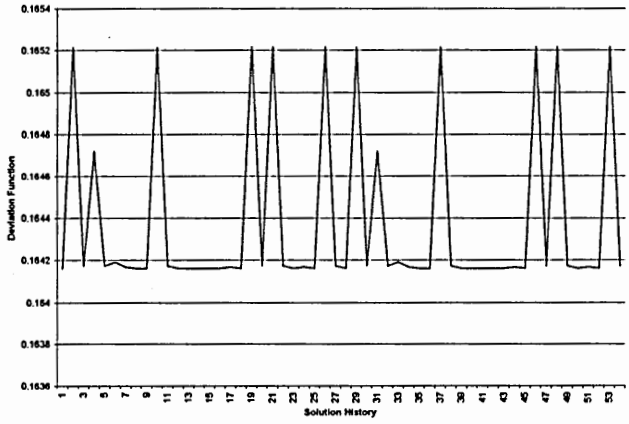


Figure 5.26 – Convergence History of GRG for One FDM Maxum

Comparison with Exhaustive Search

In order to verify that the solution acquired by the SQP solution technique provides the best possible answer to the problem, its results are compared with that of an exhaustive search. Since the problem is very complex (as explained in Section 5.2.6), the exhaustive search is run for only one machine type/quantity, namely one FDM Maxum. The comparison of the results is presented in Table 5.14.

Table 5.14 – Comparing Results of the SQP with an Exhaustive Search

	Exhaustive	*SQP*	*% difference*
ΔD_1	5	5	0
ΔD_2	35	21	40
ΔD_3	35	21	40
Average Time (hrs)	46.828	47.42	1.25
Average Cost ($)	11,597.16	11,659.32	0.533
Average Quality (mm)	0.5122	0.509	0.629
E[U(D)]	0.8360659	0.8360314	0.0041

As can be seen in the table above, the use of an exhaustive search identified a better solution in the decision space than what is reported through the use of the SQP. The implications of this result are analyzed in Section 5.3.5.

5.3.5 Analysis of the Results

A series of observations regarding the solution algorithm, the results, and the problem model itself are presented in this section as a means for further analysis of the validity of the application of the augmented PPCTM to the realm of production process design. These observations include:

- ΔD_1 is always equal to 5 parts/day
- The values of layer thickness and road width are more commonalized than what is dictated by the decision variables ΔD_2 and ΔD_3
- ΔD_2 and ΔD_3 are equal (or similar) for each machine type and quantity combination
- Different combinations of values of ΔD_X provide similar expected utilities

ΔD_1 is always equal to 5 parts/day: From Table 5.8 it is observed that for each machine type/quantity combination it is most desirable to commonalize the batch size across a range of 5 parts per day across the space of capacity. Furthermore, although these results suggest that batch size should be commonalized over a small range, there are multiple instances for many machine combinations of the same value for batch size (i.e, in Table 5.10, although ΔD_1 for a Prodigy Plus is equal to 5 parts/day, the batch size for each capacity level is 234 parts/batch; this is shown for each machine in Appendices E.1 – E.6).

While this may not seem reasonable, this occurrence is readily explained. As observed in Section 5.1.2, larger batch sizes typically provide the largest expected utility due to the lower average cost and production time. The size of each batch is limited, however, by the weeklong lead-time constraint; for small capacity requirements one cannot wait long enough to build such a large batch and still have enough time to manufacture the parts. This is corroborated by the observation that for the majority of the machines, there is a common batch size (the maximum batch size allowed) across the larger end of the space of capacity (Appendices E.1 – E.6). As soon as the threshold of violating the lead-time constraint is reached, a new batch size must be selected. Once this threshold has been surpassed, the optimum batch size for a capacity D_2 is no longer sufficient for a capacity D_1 where $D_1 < D_2$. Thus, once a certain capacity requirement is reached that is below the lead-time constraint threshold, it is infeasible to have any commonality in batch size. As such, the solution algorithm selects the smallest possible range of commonalization. This should not be considered to be a weakness of the augmented PPCTM. If a space of capacity was chosen that did not cross each machine's threshold, batch size could be easily commonalized.

The values of layer thickness and road width are more commonalized than what is dictated by the decision variables ΔD_2 and ΔD_3. From Table 5.8 it is observed that for one FDM Maxum, the values of road width and layer thickness should be commonalized across a range of capacity of 21 parts per day. It is observed in Table 5.9, however, that values of road width are actually commonalized across ranges of up to 80 parts per day. This is also observed for each machine type in Appendices E.1 – Appendices E.6.

The explanation for this occurrence is similar to that which was provided above. For smaller capacity needs, the production process requires the freedom to be able to alter the layer thickness and road width in order to meet the weeklong lead-time constraint. Large commonality ranges for these two process parameters would prevent the production process from being able to manufacture enough parts to meet the demand. One should not frown at the extra commonalization found in these results; the additional setup penalties that are typically incurred are now absent.

ΔD_2 and ΔD_3 are equal (or similar) for each machine type and quantity combination: This observation, from Table 5.8, provides another opportunity to verify the results presented. For each machine type / quantity combination, the values of ΔD_2 and ΔD_3 are equal (i.e., for one Prodigy Plus, both ΔD_2 and ΔD_3 are equal to 44 parts per day). This is explained by reinforcing the importance of the presence of setup penalties in the modeling of this example (Section 5.2.2). It is assumed that without the setup penalties present in this example problem, there would be no reason for commonalization (in terms of maximizing the expected utility of the space of capacity). These penalties are accumulated for each capacity requirement in the space in which a process parameter is changed. In an effort to minimize these penalties, the solution obtained by the algorithm is to make the ranges for the commonalization of road width and layer thickness identical. As a result, the objectives penalized a fewer number of times. This fact also offers further explanation of the previous observation: smaller ranges of commonality are chosen than that which is necessary in order to match another parameter's range of commonalization and thus minimize the total number of setup penalties for the production process family.

The solution algorithm never produces an expected utility that is an overachievement of the goal (i.e, d^+ is always equal to zero): This observation provides another manner in which to verify the results presented above. Through the formulation of the individual utility functions for each objective, it is impossible for the expected utility to ever exceed a value of 1. This is expected, as this utility value represents the ideal, and often unattainable, goal. This observation also ensures that a vital constraint of the u-cDSP is met: $d^- \cdot d^+ = 0$.

Only one machine type/quantity was chosen for the entire space of capacity: At first glance, this seems like an error. How can the most expensive machine, the FDM Maxum, be the best choice for low capacity requirements? Through careful observation of Tables 5.9 and 5.10 (or in Appendix F), an explanation is found. As mentioned in Section 5.3.3, the expected tradeoff in machine purchase cost is negated by the extra costs associated with operating the slower machines. As such, the cost of producing 120 parts per day with one Prodigy Plus ($7621.04, Table 5.10) is only slightly cheaper than with one FDM Maxum ($8765.42, Table 5.9). The FDM Maxum's higher scan speed provides a smaller build time (25.31 hrs. versus Prodigy Plus's 49.05 hrs.), and as a result, it is able to take the time to build a higher quality part (0.34 mm. versus Prodigy Plus's 0.504 mm.). As a result of all of this, the FDM Maxum is clearly the best choice for the entire space of capacity; it simply offers more *machine* for its price. It should be noted that this is not a weakness of the augmented PPCTM; it is simply a facet of the modeling of the example problem. The extreme bounds of this phase of the multistage decision were not reached; two FDM Maxums proved to be a poor choice for this space of capacity.

Different combinations of values of ΔD_X provide similar expected utilities: Through observation of the results of the application of the SQP presented in Table 5.12 (and Appendix D), one can observe that there is no major change in expected utility amongst machines and different values for each mode for managing variety, ΔD_X. Further proof of this facet of the results is provided in Table 5.15, a small portion of the iteration history of applying the SQP for one FDM Maxum (from Appendix D.1).

Table 5.15 – Similar Expected Utilities for Different Ranges of Modes for Managing Variety in Hearing Aid Example Problem

ΔD_2	ΔD_2	E[U(D)]
50	150	0.835839
27.6	78.7	0.835857
22.87	22.87	0.836024
5	33.015	0.835171
80	85	0.835855
65	70.37	0.835947
34.74	86.67	0.836026
25	350	0.833468

Here in lies a major limitation of the use of an expected utility function for this specific example problem. The formulation of the expected utility function (Table 5.3) is broad enough to capture the extremely different capabilities and properties of each machine. This results, however, in a function that is insensitive to small changes in decision parameters. The expected utility function used for this example problem is formulated in order to insure that each machine is capable of producing a feasible result – if the maximum cost allowed by the utility function were made any lower, the Prodigy Plus would be considered. It is therefore recommended that a designer's preferences be clearly identified, and be as scoped as possible for the application of the augmented PPCTM to this type of problem.

It is noted that the use of the u-cDSP is not a limitation of the augmented PPCTM. The expected utility function's insensitivity to small parameter changes is a result of the model itself. The robustness of the hearing aid example model is documented in (Williams et al., 2003).

Closing Observations

Closing observations regarding this example problem are offered in this section in order to provide deeper insight into the meaning of the results received through the application of the augmented PPCTM. These observations are centered on four themes:

- The validity of the use of SQP
- The use of a multi-attribute utility function
- The absence of risk and uncertainty in the problem model
- The overall usefulness of applying the augmented PPCTM to the design of a production process.

The validity of the use of SQP: There exists a major effort to verify and validate the results obtained via the SQP solution technique in Section 5.3.4. Different starting points are attempted and two different solution techniques are used to assist in the validation of the results. The use of the exhaustive search, however, provides a better solution to the problem than the SQP (Table 5.14).

While the results reported in Table 5.14 suggest that the use of the SQP technique is not appropriate for this specific example, it does not suggest that it is inappropriate for general use in other applications of the augmented PPCTM. The use of an exhaustive search for this example is extremely computationally expensive. 15,575 iterations of the multi-stage u-cDSP (Figure 5.19) are required to solve this example with an exhaustive

search. Far less iteration is required in the implementation of the SQP (around 90, Figure 5.25). As a result, the use of a solution technique that is capable of altering its input guesses based on output results is an important avenue of research for these types of problems. The question arises, however, why did the SQP fail to present the ideal solution for this particular problem?

For this particular problem, the largest expected utilities are found when the decision variables ΔD_2 and ΔD_3 are identical (as noted earlier, and shown in Table 5.8). As such, there are small peaks in values of expected utility in the solution space when both decision parameters are equal. These small peaks do not provide accurate gradient information on which the SQP can search by. When coupled with the relatively insignificant difference between expected utilities of different ranges of modes (as noted earlier, and presented in Table 5.14), the implementation of the SQP is simply unable to detect the presence of a solution far from its starting point that provides a higher utility.

It should be noted that the use of a gradient-based solution technique is a form of art, rather than a science. Many different attempts are made in the solution of this problem in order to find an appropriate set of SQP application conditions (Table 5.7) that make the technique somewhat useful. Each decision variable is normalized upon input, and the resultant expected utility is multiplied by a factor of 1000 in order to magnify the effects of changing the input values. In this sense, the brute-force style of an exhaustive search is more desirable despite its computational intensity.

The use of a multi-attribute utility function: In Table 5.15, the insensitivity of the expected utility value to changes in decision parameters is observed. As discussed earlier, this is a result of formulating the individual utility functions across a large

spectrum of preferences. The fact that the results presented must be compared to the fourth significant digit in order to make a decision, suggests that the results are not reliable enough to be certain that the best decision has been reached when using a gradient-based solution technique. Specifically, the small differences between the values of expected utility for each level of capacity amongst each machine type/quantity combination prevent a designer from being certain that the ranges of each mode are indeed ideal. Since there exist larger differences in expected utility between each machine type and quantity, an engineer can be confident in this portion of the decision (Decision 2, Figure 5.20).

This concern regarding the reliability of the results can be solved in one of two ways: the use of an exhausted search solution technique to ensure that the best solution has been found, or the reformulation of the individual utility functions of the multi-stage u-cDSP in order to scope the preferences of the engineer.

The absence of risk and uncertainty of the problem model: The absence of risk and uncertainty in the modeling of this example provides a large limitation on the usefulness of the method in an industrial setting. Specifically, the inability of the problem model to capture the possibility of machine down time (usually due to build crashes) prevents an engineer from trusting the results. This is especially relevant due to the fact that the ideal set of parameters for most capacity levels have the process pushing the limit of the weeklong lead-time constraint. In order to make this research effort more applicable to an industry setting, the risk and uncertainty involved with rapid manufacturing must be captured in the problem model.

The overall usefulness of applying the augmented PPCTM to the design of a production process: The case for establishing the usefulness of the augmented PPCTM in the design of a production process is presented by answering two questions:

1. *Is the development of a production process platform a good approach for this problem?*
2. *Is the use of the PPCTM a good approach for the solution of this problem?*

In order to answer these questions, the values of each objective are compared between the results from the development of a production process platform (using the PPCTM), the result of having no commonality in process parameters for each capacity requirement, and the result of making all process parameters common across the space of capacity. This comparison is provided in Table 5.16.

Table 5.16 – Comparison of a) Augmented PPCTM Process Platform, b) No Commonality, and c) Strict Commonality of Process Parameters

	t_{avg} (hrs)	C_{avg} ($)	Q_{avg} (mm)	$E[U(D)]$
Augmented PPCTM Result	47.42	11,659.32	0.509	0.8360
No Commonality ($\Delta C_X = 1$)	48.16	11,734.33	0.507	0.835
Pure Commonality ($\Delta C_X = 880$)	43.52	12,681.23	0.756	0.720

Although the benefit for this specific problem is not significant, it is evident that the development of a production process platform improves all three of the manufacturer's objectives (thus answering the first question).

The second question presented above (the usefulness of the PPCTM in the creation of a process platform) is answered by identifying the number of decisions required to develop the platform. In order to determine the ideal process parameters for each capacity variant in the space of capacity (Figure 5.6), an engineer would have to run the u-cDSP of "Decision 0" (Figure 5.13) 880 times (once for each variant in the space). If

an engineer desires to develop a process platform for this problem, "Decision 0" would only be solved at each bound of the sub-spaces (as explained in Section 5.2.6). Although the problem must be solved numerous times for different guesses for appropriate ranges of the modes for managing variety (the number of times is dependent on the solution technique used), each design objective would be improved as a result of the development of the production process platform (Table 5.15).

5.4 THESIS ROADMAP: A LOOK BACK AND A LOOK AHEAD

The reasons for, and a description of, the augmented PPCTM are presented in the first three chapters of this thesis. The application of the augmented PPCTM to an example of product platform design is presented in Chapter 4. In this fifth chapter, the augmented PPCTM is applied to a new, but similar, domain: process platform design. Specifically, the augmented PPCTM is used to develop a process platform for the manufacturing of customizable hearing aid shells. The purpose of this chapter is the further establishment of empirical structural and empirical performance validity of the proposed augmented PPCTM.

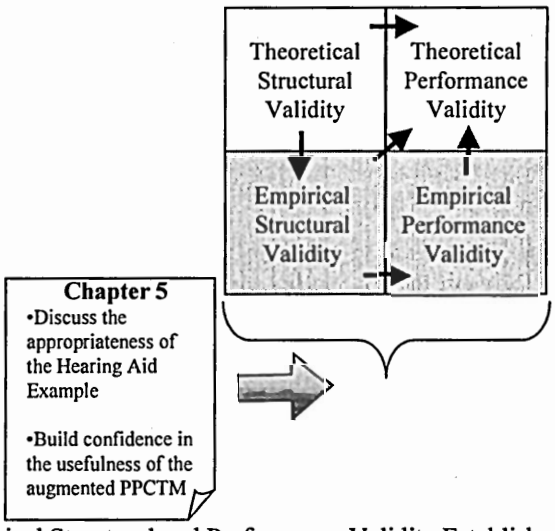


Figure 5.27 – Empirical Structural and Performance Validity Established in Chapter 5

The example problem presented in this chapter is a germane and appropriate problem for analysis. Although rapid manufacturing is a specialized form of production, it has many caveats in common with traditional manufacturing techniques. Conflicting objectives, numerous process parameters, and discrete process selections are all a part of general process design, and are successfully handled with the application of the augmented PPCTM to this example problem. The production of customizable hearing aid shells is also relevant to current research in industry – it is based on products currently offered through a joint effort between Siemens and Phonak.

The establishment of empirical performance validation involves proving that the application of the augmented PPCTM to this example produces valid results. A presentation and critical analysis of the results was discussed in Section 5.3. In this analysis, several questions regarding the validity of the results were raised and then subsequently explained (Section 5.3.5).

The purpose of this example problem is to show that the augmented PPCTM can be applied to the realm of process platform design (Research Question #3). From the comparison of the results of the application of the augmented PPCTM to those results without (Table 5.15), it is concluded that its application to production process design is a success.

This example problem was also chosen in order to test the ability of the augmented PPCTM to handle multiple objectives (Research Question #2). Based on the results presented in Section 5.3, it is evident that the augmented PPCTM succeeded in this as well. The infusion of the utility-based compromise Decision Support Problem provides a mathematically rigorous decision model in which a designer's preferences are captured

and quantified. It should be noted, however, that the use of an overall expected utility function in this example did provide some insensitivity to changes in decision parameters.

In the next, and final, chapter of this thesis a summary of the work is presented. A review of the validation strategy alongside an establishment of theoretical performance validity is given. Finally, a summary of contributions and suggestions for future work are presented.

CHAPTER VI

A LOOK BACK AND A LOOK AHEAD: CLOSING REMARKS

In this thesis, augmentations to the Product Platform Constructal Theory Method (PPCTM) are presented. These augmentations are focused towards providing a designer the ability to handle non-uniform demand in both product and process platform design. A primary contribution of the completion of this work is the application of the augmented PPCTM to the realms of product and process design through two example problems – the product platform design of a line of customizable pressure vessels, and the process platform design of the production of customizable hearing aid shells.

In this chapter the development and implementation of this approach is brought to a close (Figure 6.1). Closure is obtained through the review of the work with emphasis on its validation and verification in Section 6.1. A presentation of the achievements and contributions detailed in this thesis are presented in Section 6.2. Next, in Section 6.3, a critical analysis of the work is discussed along with a review of its limitations. Potential avenues of future work are recommended in Section 6.4. This chapter concludes with the author's closing remarks in Section 6.5.

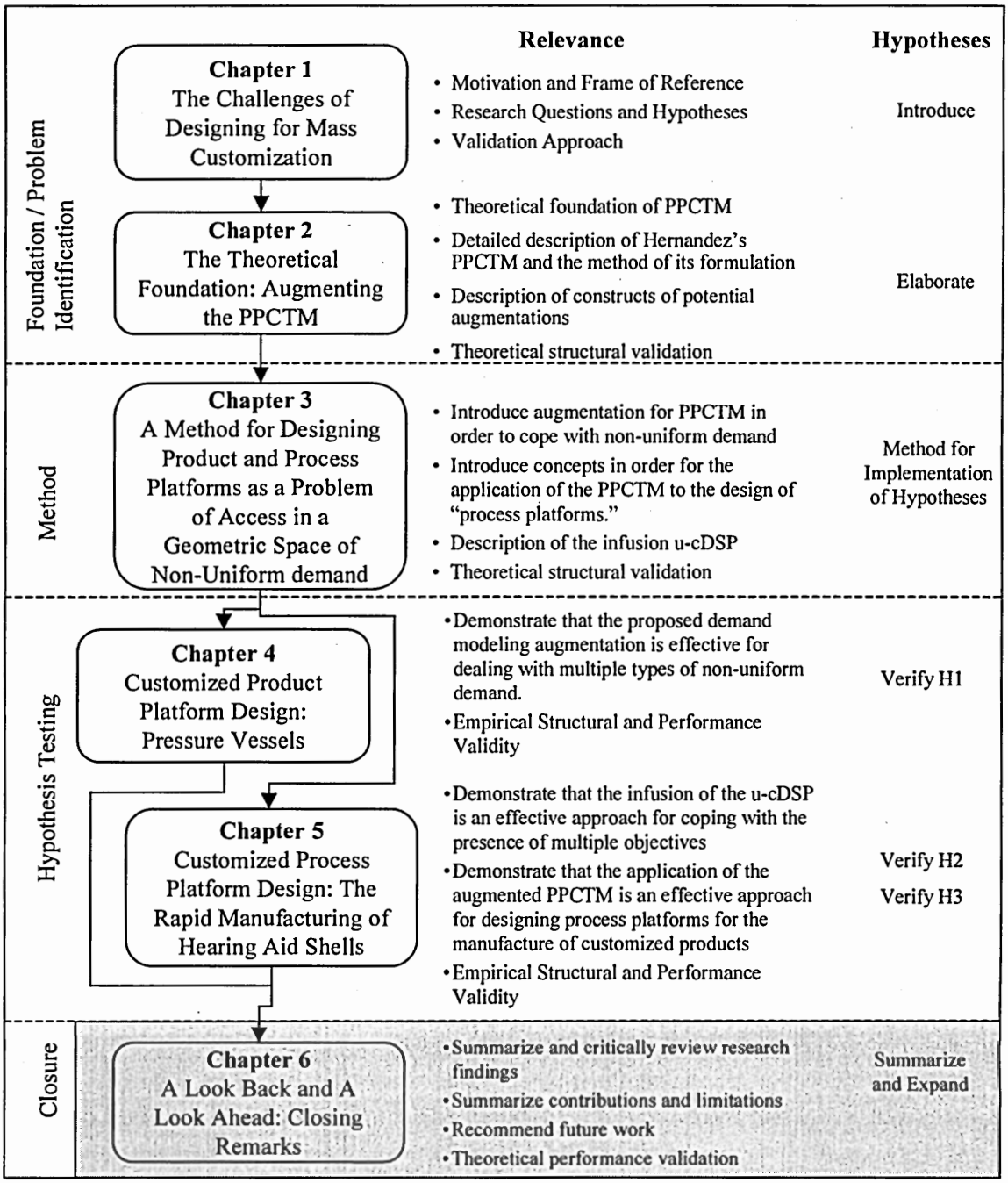


Figure 6.1 – The Role of Chapter 6 in this Thesis

6.1 A LOOK BACK: A SUMMARY OF WHAT HAS BEEN PRESENTED AND ITS CONTRIBUTION TO VALIDATION

In this section, the validation square is used to demonstrate that the augmentations of the PPCTM proposed in response to the research questions of Chapter 1 are valid. This validation strategy is also used as a vehicle to review the work presented thus far. In Section 6.1.1 the research questions and accompanying hypotheses are presented. In Section 6.1.2, the work is reviewed and validated in the context of each portion of the validation square.

6.1.1 Revisiting the Research Questions and Hypotheses

Research involves posing appropriate questions, formulating answers to these questions as a set of hypotheses, and demonstrating that the proposed answers are valid. For this work, the validation task involves building confidence in the usefulness of the proposed augmentations for meeting the challenges posed by the research questions. Before the specifics of the validation strategy are reviewed and renewed, the research questions presented in Chapter 1 are revisited.

As discussed in Chapter 1, the primary research question posed in this thesis is:

How can the Product Platform Constructal Theory Method be augmented to synthesize multiple modes of managing customization in order to create product and process platforms for customizable products in a market of non-uniform demand?

This primary research direction is divided into three separate research questions and associated hypotheses:

Research Question 1: How can the Product Platform Constructal Theory Method be augmented in order to cope the development of product platforms in a market of non-uniform demand?
Hypothesis 1: Modeling the demand as a property of the geometric market space, a designer can successfully solve the non-uniform optimal access problem through the discretization of the space.

Research Question 2: How can the Product Platform Constructal Theory Method be augmented in order to enable a designer to cope with the presence of multiple objectives?
Hypothesis 2: The infusion of the utility-based compromise Decision Support Problem will provide a rigorous construct in which a designer can make decisions with multiple, conflicted, and coupled objectives.

Research Question 3: How can the Product Platform Constructal Theory Method be successfully applied to the domain of production process development for customized products in a market of non-uniform demand?
Hypothesis 3: The generalized application of the PPCTM (and the augmentations proposed in Hypotheses 1 and 2) can be applied to assist a designer with the development of a "process platform" for the manufacture of customized products.

Answering each of these research questions involves the validation of the corresponding hypotheses. Aspects of this verification are presented throughout this thesis (as presented in Figure 1.8) and are brought together using the validation square (as presented in Section 1.4) in the next section.

6.1.2 Testing the Validity of the Proposed Augmented Method

An overview of the validation strategy pursued in this thesis is given in Section 1.4. The strategy is formulated according to the validation square (Pederson et al., 2000) as shown in Figure 1.7. Figure 6.2, a reincarnation of Figure 1.8, illustrates the manner in which the work of this thesis has been focused towards its validation.

Theoretical Structural Validation	
Review of product platform design techniques, thorough description of Hernandez's PPCTM (which serves as a foundation for this work), and a discussion of the constructs to be included to deal with non-uniform demand and the presence of multiple objectives.	Chapter 2
Presentation of step-by-step method and an implementation flow-chart for this approach. Explicitly stating the augmentations to be added the PPCTM.	Chapter 3
Empirical Structural Validation	
Discuss the appropriateness of the pressure vessel example problem: - Document that the abstracted ideas of the constructs presented are relevant to the application of the pressure vessel example problem. - Document that the pressure vessel example problem is representative of an actual problem for which the approach is intended. - Document that the data associated with the pressure vessel example problem can support a conclusion or conclusions for Hypothesis 1.	Chapter 3, Chapter 4
Discuss the appropriateness of the hearing aid shell example problem: - Document that the abstracted ideas of the constructs presented are relevant to the application of the hearing aid shell example problem. - Document that the hearing aid shell problem is representative of an actual problem for which the approach is intended. - Document that the data associated with the hearing aid shell example problem can support a conclusion or conclusions for Hypotheses 2 and 3.	Chapter 3, Chapter 5
Empirical Performance Validation	
Build confidence in the usefulness of the method using the pressure vessel and hearing aid shell example problems.	Chapters 4 and 5
Theoretical Performance Validation	
Build confidence in the generality of the approach or in the usefulness of the approach beyond the specific example problems.	Chapter 6

Figure 6.2 – Validation Strategy Related to Thesis Format

Three of the four components of the validation square, namely theoretical structural validity, empirical structural validity, and empirical performance validity, are addressed throughout the presentation of the previous chapters of this thesis. In this section these arguments will be briefly summarized in order to bring them together. The final component of the validation strategy, theoretical performance validity, will also be addressed.

Theoretical Structural Validity

For this thesis, theoretical structural validity involves the consideration of advantages, limitations, and accepted domains of application for each of the constructs constituting the augmentations of the PPCTM. This is accomplished through a literature review and a critical review of each of the building blocks considered. There are two major decisions involved with accepting the theoretical structural validity of this work; they are explained below.

1) Are the constructs themselves valid? Have they been cited in literature before?

This question is addressed in Chapter 2 where the theoretical foundation of this work is established. Some highlights of this presentation include:

- The core focus in this work is the augmentation of the Product Platform Constructal Theory Method. It is critically reviewed in Section 2.1. Specifically, an overview of its building blocks, hierarchic systems theory and constructal theory, is provided in Section 2.1.1. The six steps of the PPCTM are then detailed in Section 2.1.2. Finally a critical analysis of the method is provided in Section 2.1.3, complete with an itemized list of its limitations. This section also provides a series of citations documenting its previous domains of application.
- The theoretical structural validity of the utility-based compromise Decision Support Problem is addressed in Section 2.2. Its theoretic underpinnings, the compromise Decision Support Problem and utility theory, are summarized in Sections 2.2.1 and 2.2.2. The four-step formulation method of the u-cDSP is then detailed in Section 2.2.3. A critical analysis of the u-cDSP, as well as a series of citations of its previous applications, is provided in Section 2.2.4.

2) Is the resulting combination of these constructs method valid? Is the combination consistent with the constructs themselves?

This question is discussed in Chapter 3 where the augmented Product Platform Constructal Theory Method is presented. Some salient points of this discussion include:

- An overview of three methods for modeling non-uniform demand is presented in Section 3.1. Their validity is verified through the presentation of their mathematical formulations and application to the domain of product platform design.
- The application of the PPCTM to the realm of production process design is discussed in Section 3.2. Specifically, the concepts of a process platform, a space of capacity, and modes for managing process customization are developed through their definition and discussion of their need. In Section 3.2.4, the concerted application of these concepts and the augmented PPCTM to the realm of process design is conveyed.
- The infusion of the u-cDSP into the PPCTM is discussed in Section 3.3. In Section 3.3.1, specific steps of the original PPCTM are highlighted to show their resulting change due to the infusion of the u-cDSP decision construct. A critical review of the resulting infusion is addressed in Section 3.3.2. This critical analysis provides much information regarding the theoretical structural validity of the infusion of the u-cDSP into the PPCTM.
- In Section 3.4, the formulation of the augmented PPCTM is presented. Each of the six steps is detailed in Sections 3.4.1 through 3.4.6, wherein the new augmentations are highlighted.

- Finally, in Section 3.5, a review of the theoretical structural validity of the augmented PPCTM is provided. Details of the validity of infusing the u-cDSP and non-uniform demand modeling techniques, as well as the application of the PPCTM to process design, are discussed.

As a result of this portion of the validation exercise, confidence in the correctness and consistency of the constructs of the PPCTM and the u-cDSP, as well as their combination to create the augmented PPCTM, is built.

Empirical Structural Validity

Empirical structural validity involves building confidence in the appropriateness of the example problems chosen for illustrating and verifying the performance of the design method. One decision is required for the completion of this portion of the validation square: *Are the example problems are appropriate?* Three steps are suggested for answering this question:

1) Are these example problems similar to the domain of which the constructs are generally accepted?

2) Do the example problems accurately represent the real world situations they attempt to portray?

3) Can the data obtained from the example problems support a conclusion with respect to the research hypotheses?

These questions are addressed in the development of both example problems in Chapters 4 and 5.

The focus in Chapter 4 is the development of an example problem where the augmented PPCTM is used to design a product platform. Specifically, the augmented

PPCTM is applied to the design of a product family for a line of customizable pressure vessels. In Section 4.4, the empirical structural validity of this example problem is addressed. In this section it is noted that the pressure vessel example problem is appropriate:

- The pressure vessel is a classic mechanical engineering example that is simple enough to allow the reader to focus on the method itself, and yet complex enough to provide a challenge for the application of the method.
- The PPCTM has been applied to a pressure vessel in previous work (Hernandez, 2001; Carone et al., 2003).
- The u-cDSP has been previously applied in the development of product platforms (Seepersad, 2001; Seepersad et al., 2002).
- The modeling of the pressure vessel is presented in Section 4.1. Although this modeling effort is limited and based on a certain set of assumptions (listed in Section 4.1), it is still an appropriate, and often used, manner in which to approximate the design of a pressure vessel.
- The purpose of the use of the pressure vessel example problem is to test the validity of the first hypothesis. This is accomplished by applying the augmented PPCTM to several different non-uniform demand scenarios in Section 4.3. In Section 4.3.5, numerous points regarding the validity of the results are discussed.

The focus in Chapter 5 is the application of the augmented PPCTM to the realm of process design in order to provide empirical structural validity to the second and third hypotheses. Specifically, the augmented PPCTM is applied to the design of a process

platform In Section 5.4 this validation effort is summarized. Regarding the three questions presented above, the use of the hearing aid shell example problem is valid:

- While neither the PPCTM nor the u-cDSP has been applied to the domain of process design, they have been extensively applied to the realm of platform design. The similarities of these two domains are explained in Section 5.1.
- The focus in the hearing aid production example problem is on a specialized form of production involving the use of rapid prototyping technologies named rapid manufacturing. As outlined in Section 5.4, although this example problem deals with a specialized form of production, its key decisions and complexities are a result of problems that appear in all forms of "traditional" manufacturing techniques.
- The example problem is based on an industrial application - the rapid manufacturing of hearing aid shells is currently in production through collaboration between Siemens and Phonak. The data of the Stratasys Fused Deposition Modeling machines, as presented in Section 5.1.2, is based on information acquired directly from the manufacturer.
- The limitations and assumptions of this example are presented in Section 5.1. Although the problem is extremely bounded, this must be done in order for it to be adequately handled.
- The results of the application of the augmented PPCTM are reported in Section 5.3. The justification of their validity is provided in Sections 5.3.4 and 5.3.5. The success of the application of the augmented PPCTM to process design is noted

alongside a discussion of its ability to handle multiple objectives, thus validating the second and third hypotheses.

With the answering of each of these three questions, the augmentations proposed in Hypotheses 1, 2, and 3 (modeling of non-uniform demand, the infusion of the u-cDSP, and the application to production process design, respectively) are declared empirically structurally valid.

Empirical Performance Validity

The establishment of empirical performance validity includes building confidence in the usefulness of the augmented method. There are two steps involved with building confidence in the augmentations:

1) Decide that the method is useful for the example problems.

2) Decide that the usefulness is a direct result of the application of the method.

These decisions are addressed in Sections 4.4 and 5.4 for the pressure vessel and hearing aid example problems respectively.

- The results of the application of the augmented PPCTM to the pressure vessel example problem are presented in Section 4.3.4. These results are deemed useful through their comparison with two scenarios: a product platform with no commonality, and a product platform with extreme commonality. As shown in this section, the use of the augmented PPCTM provided a higher profit than either of the two scenarios.
- The results of the application of the augmented PPCTM to the hearing aid example problem are presented in Section 5.3.3. These results are deemed useful through their comparison with two scenarios: a process platform with no commonality, and

a process platform with extreme commonality (Section 5.3.5). As shown in this section, the use of the augmented PPCTM provided a higher expected utility than either of the two scenarios.

Since the augmentations proposed in Hypotheses 1, 2, and 3 are demonstrated to be useful in terms of designing product and process families for customized products in markets of non-uniform demand with multiple objectives, it is concluded that the augmented PPCTM is valid in terms of empirical performance, for the chosen example problems.

Theoretical Performance Validity

Theoretical performance validity involves building confidence in the generality of the augmented method and accepting it as useful beyond the example problems. This involves showing that the example problems are representative of a general class of problems and that the augmented method is useful for these problems.

The general class of problems for which the constructs of this augmented PPCTM are valid can be defined by the following characteristics:

For product platform development:

- Design problems involving multiple conflicting objectives
- Design problems involving a market of non-uniform demand
- Product platform development problems involving dimensional customization, modularity, and parametric scaling
- Product platform development in which the products are functionally similar, but with different performance requirements

- Product platform development problems involving a distinct tradeoff for commonalizing a design parameter for the sake of performance
- No uncertainty in the design information, design objectives, or goals is present

For process platform development:

- Design problems involving multiple conflicting objectives
- Design problems involving a non-uniform levels of capacity requirement
- Process platform development problems involving parametric scaling of process parameters
- Process platform development in which one production workstation is analyzed
- Process platform development in which the production involves one type of product that is customized through the use of the workstation being analyzed
- Process platform development problems involving a distinct tradeoff for commonalizing a design parameter for the sake of performance
- No uncertainty in the design information, design objectives, or goals is present

For the implementation of the utility-based Compromise Decision Support Problem:

- Decision-makers know what they want; there is no uncertainty involved with their preferences
- Decision-makers are rational as defined by the axioms of utility theory

It is noted that the augmented PPCTM provides a framework for a designer to deal with multiple (more than two) varied design parameters, as well as the ability to deal with uncertainty in the design parameters. These two facets are not explicitly proven in this work, and are therefore not formally claimed here.

As part of the requirements of establishing empirical structural validity, it has been shown that the example problems are representative of a class of general problems with the preceding characteristics. In order to establish empirical performance validity, it has been shown that the methods are useful for the example problems. By induction, it is argued here that the methods are useful for a general class of problems identified by the preceding characteristics.

As a result of this validation strategy, each portion of the validation square has been reviewed. It is therefore asserted that the primary research question has been answered. *The Product Platform Constructal Theory Method has been augmented in order to synthesize multiple modes of managing customization in the development of both product and process platforms in a market of non-uniform demand. This has been accomplished through the synthesis of mathematical demand modeling techniques, the infusion of the utility-based compromise Decision Support Problem to handle multiple objectives, and the application of the method to the realm of production process design.*

An overview of the contributions of this work is now provided in the following section.

6.2 ACHIEVEMENTS AND CONTRIBUTIONS

As indicated in Chapter 1, the primary research question for this thesis is:

> *How can the Product Platform Constructal Theory Method be augmented to synthesize multiple modes for managing variety in order to create product and process platforms for customizable products in a market of non-uniform demand?*

Inherent in this research question is the author's desire to accomplish three goals with the completion of this thesis:

i) to provide a designer using the PPCTM the ability to design a product or process platform for customizable products in the face of any type of market of non-uniform demand

ii) to provide a designer using the PPCTM the ability to handle multiple conflicting objectives

iii) to provide evidence that the PPCTM can be successfully applied in the domain of production process design.

With these three goals established, a series of contributions have been made as a result.

First, *with respect to the Product Platform Constructal Theory Method specifically*, many key contributions are made. The infusion of the utility-based compromise Decision Support Problem provides designers with the ability to handle multiple conflicting objectives in platform design. The infusion of the u-cDSP also takes the emphasis on optimization away from the PPCTM, thus allowing a designer to select a "satisficing" solution. Multiple, different techniques for modeling non-uniform demand are used in the development of platforms for customizable products. The inclusion of these methods, and the incorporation of demand modeling as a specific step of the augmented PPCTM, alleviates the previous versions dependence on the assumption of uniform demand across a market. Finally, the augmented PPCTM is successfully applied to the domain of production process design. This provides credibility to the design method by showing that it has strength outside of the bounds it was originally intended for.

With respect to the field of product platform design, a product platform for a line of customizable pressure vessels is developed using the augmented PPCTM. The use of this method provides a designer with the ability to synthesize multiple modes for managing

product customization in a hierarchic manner. Specifically, dimensional customization, parametric scaling, and modularity are used in concert to develop a product platform with the objective of maximizing profit across the entire product line. This augmented method uses the inherent tradeoffs between commonality and performance in order to determine the most appropriate range of commonality for each mode of managing product variety. The augmentations of the PPCTM presented in this thesis provide a designer the ability to appropriately model the demand of the market space in order to make more educated decisions regarding the design of the product platform. Finally, as a result of this specific augmentation, a list of observations and lessons learned regarding the relationships between product platform extent and the demand for these products are presented. It is the author's hope that these lessons will be of use to the product platform design community and their efforts in capturing adequate information for the development of product platforms.

Finally, *with respect to the design of production processes*, a number of new concepts are introduced in this thesis. The definition of, and reasons for, a process platform is introduced. This concept of a process platform is born from the need to shorten design time and to relieve penalties associated with the setup required for the production of different parts and of different capacities. Furthermore, the concept of a space of capacity – the abstraction of capacity need as a problem of optimal access in a geometric space - is generated in order to apply the concepts of the PPCTM to this new domain. Along a similar note, the concept of modes for managing process customization is created in order to allow designers to visualize process parameters as means of moving from one process design to the next with minimal resistance. Much like (Hopp et al.,

2001), it is the author's hope that the application of decision-based design to the realm of process design adds a hint of "science" to the current methods of manufacturing systems design.

This list of contributions is a result of searching for an answer to the overall research question. With these contributions presented, a critical analysis of the work is presented in the following section.

6.3 A CRITICAL ANALYSIS AND REVIEW OF THE LIMITATIONS OF THIS RESEARCH

While the goals established at the beginning of this thesis are close to being accomplished, there are a number of limitations of this work that must first be addressed. A large portion of this critical analysis is presented throughout this thesis, specifically in the application of the augmented PPCTM to the two example problems in Chapters 4 and 5. These criticisms are summarized again here. It should be noted that many of the limitations of this research will be transformed into avenues for future work in the following section.

Obviously, several limitations of this research are derived from the series of assumptions listed at the beginning of the description of each of the example problems:

- *A fixed market space*: For the use of the augmented PPCTM in the development of product platforms, such as the pressure vessel example in Chapter 4, it must be assumed that the market space will not change over a period of time. This not only applies to the bounds of the market space, but also to the demand scenario initially modeled. This assumption limits the application of the augmented PPCTM in two ways:

- Only a "snapshot" of the current market can be analyzed; any changes to this market will require a completely new series of analyses.
- One key purpose of the development of a product platform is the resulting design's ability to be easily modified into a new product as a result of market change. Whether through a change in customer requirement, or through a change in the vision of the manufacturing enterprise, a product platform should have the "openness" to evolve into a new series of product through technology infusion or a change in function. This current limitation of the augmented PPCTM prevents a designer to make decisions with the necessary foresight for these types of designs.

It should be noted that this assumption is not present in the formulation of the augmented PPCTM for the development of a process platform. The nature of ever-changing requirements for capacity of a production process is the key driver behind the concept of a process platform.

- *Scalable product components*: For the application of the augmented PPCTM to the development of both product and process platforms, only scalable product components are considered.
  - For product platform design, this assumption relates to the inability to model the manufacturer's intent to offer variety through functionality. A common limitation of product platform design techniques, this assumption effectively eliminates the ability of the designer to use a series of often used modes for managing product customization.

- This assumption's impact on process design relates to the limitation of the augmented PPCTM to model the production of more than one type of product. For the hearing aid example problem it is a fortunate coincidence that the process being considered automatically produces customized parts; each unique part does not need its own batch or its own unique production process. While the utility-based compromise Decision Support Problem has been used to design a production process for the rapid manufacturing of a series of different product variants (Williams et al., 2003), this has not been accomplished for a series of demand requirements.

- *Complete and certain information*: A major assumption of the application of the augmented PPCTM is the absence of uncertainty in the design parameters, objectives, or goals. In the effort to model non-uniform demand, this assumption provides a major weakness to the claim that the augmented method can support uncertainty in demand. Conceptually, the infusion of the utility-based compromise Decision Support Problem into the PPCTM should provide a designer with the framework in which to model such uncertainty, it has yet to be specifically proven that this can be accomplished. This is a reoccurring weakness of the PPCTM (Hernandez, 2001), and is a strong direction for future work.

Expanding on these assumptions provides more insight into the limitations of the presented augmentations to the PPCTM.

- Although it is claimed that the augmented PPCTM can successfully synthesize multiple modes of managing variety, only a few are modeled in the example problems. Specifically, dimensional customization, parametric scaling, and

modular combination are the only modes for managing product customization explored. In order to fully accept the generality of this method, further modes should be explored (i.e., functional change, conventional modularity, etc.). It should also be noted that a tradeoff between performance and commonality must be included into the model of the product platform for each mode of managing product customization to be used. Without a specific reason for commonality, a mode will be completely ignored by the application of this method.

- Further exploring the limitations of the use of utility theory, it is noted again here that its use might not be entirely warranted in the example problems of this thesis. In these example problems, the utility-based compromise Decision Support Problem is used in order to quantitatively capture the preferences of the multiple conflicting objectives. One of the key strengths of utility theory, its ability to model uncertainty and risk, fails to be taken advantage of in this work. As mentioned above, although the framework is in place for future work, the effectiveness of its presence has yet to be fully explored. Another note of concern regarding the use of utility theory is its general usefulness. It has been suggested before that the application of utility theory beyond the confines of research might not be entirely realistic (Fernández, 2002). Its dependence on the rationality decision makers as well as their ability to effectively quantify their preferences consistently is a major weakness of the construct. In Section 5.3.5 the difficulty in using the u-cDSP to develop an effective expected utility theory for a large range of design variables is reviewed in order to provide evidence of the difficulty of preference modeling.

- With respect to the assumptions of the example problems in general, a limitation of this work lies in its limited approximations of these problems for their adaptation to research. Each example problem is modeled as effectively as possible while keeping the size of the resultant approximation in check. The lack of uncertainty in the problems provides the problem with convenient bounds.

While this list of limitations may seem lengthy, it is the author's opinion that they do not detract from the overall utility and validity of the augmented PPCTM. Some assumptions must be in place in order for the research to be able to be effectively handled. Also, several of these limitations provide an avenue for the free exploration of the three core tasks in the completion of this research. This list of limitations is combined with a few of the author's aspirations in order to create a series of avenues of potential future work in the next section.

6.4 A LOOK AHEAD: FUTURE WORK

One exciting aspect of performing research is that its conclusion one realizes that there is something new to explore. These efforts in augmenting the Product Platform Constructal Theory Method prove to be no different. In this section a series of potential avenues for future work are discussed.

- Distributed engineering: In order to further test the theoretical performance validity of the augmented PPCTM, its application into a distributed engineering framework, such as the System Realization Laboratory's X-DPR (Panchal et al., 2002) would be an interesting line of research. As design teams become more disparate, and design becomes more of a function of the product's supply chain,

the opportunity to use this augmented method might prove to be useful. As presented in Appendices A and C, the analysis code for the implementation of the two example problems of this thesis are written in the JAVA programming language, which would greatly expedite the translation of the method onto the web-based framework.

- Product Platform Constructal Theory Decision Software: Along the same research lines, an implementation of the generic PPCTM framework as a decision software template would also prove useful. Currently, new analysis code is written for each separate example problem. Due to the object-oriented nature of the JAVA programming language, a few functions are shared between the two example problems. A software package allowing a designer to simply input an approximation of the geometric design space and an analysis code packet could prove to be extremely useful and helpful to future designers.
- The use of the PPCTM to explore new markets: The inability to model more than a single instantiation of a market is listed as a major limitation of the augmented PPCTM. The use of the augmented PPCTM to explore new markets is an exciting area of future work. The PPCTM's synthesis of multiple modes of managing product customization could be easily applied to an analysis of the profitability of moving into a new market. This research could also include the exploration of technology infusion techniques as potential modes for managing customization. This exploration would also alleviate the current limitation involved with ignoring the opportunity of offering functional variety. A major advantage of developing product platforms lie in their ability to evolve in coordination with changes in

market. Future work along these concepts could provide a designer with the power design a more effective product platform.

- The infusion of robust design principles into the PPCTM: This is perhaps the most significant proposal for future work. The infusion of robust design concepts into the PPCTM would provide a designer the ability to effectively cope with changes in demand, changes in the market, and the uncertainty involved with the design parameters and objectives. The resulting product and process platforms would be more immune to the limitations currently put in place through the use of the PPCTM. It is the author's opinion that this is the most important direction for future work in this area.

With the validation, contributions, limitations, and ideas for future work presented, it is now time to offer some closing remarks.

6.5 CLOSING REMARKS

In this section, I present a few remarks to assist in the "deep-thinking" of the reader. These closing comments contain a few questions, a few answers, a few rhetorical thoughts, and a few lasting impressions.

At the conclusion of the development of a new design method, there are two frequently asked questions posed by the reader:

1) *What are the reasons for selecting these specific constructs?*
2) *What is the designer's role in the application of this method?*

The answer of the first question must be presented in two separate bullet-points:

- *Why constructal theory? Why the Product Platform Constructal Theory Method?* As presented in Section 1.2.4, the PPCTM is the only product platform design method that provides the designer the power to synthesize multiple modes of managing product customization and to inherently cope with the tradeoffs involved with commonality and system performance. For these reasons alone, the PPCTM is worthy of research. The abstraction of the fundamental concepts of constructal theory is a novel and effective manner in which to tackle the problems of product platform design. As shown in this thesis through the application of the PPCTM in the domain of production process design, the fundamental constructs of this design method provide hope for future implementation across a broad spectrum of disciplines. I note however, that while the use of the fundamental constructs of constructal theory provide a novel problem formulation technique; I believe that there is currently too much emphasis on the term itself. When the details of this research are first presented, the audience is typically lost in the difficult concepts of constructal theory. Technically, in the application of the PPCTM, one is not applying true "constructal theory." As presented in Section 2.1.1, constructal theory involves the sequential optimization of the smallest to largest geometric spaces – this is not a part of the PPCTM. What *is* borrowed from constructal theory, and what is so useful and important in the PPCTM, is the abstraction of the concepts to hierarchically synthesize multiple modes of product customization in the development of a product platform. What is most impressive about the PPCTM is not that it is based on constructal theory; instead what is most impressive to me is that it is methodical and mathematically rigorous method in

which one can determine the appropriate range of a series of product customization techniques across a product family. As of the writing of this thesis, the acronym "PPCTM" is not yet set in stone. It is my recommendation to the owner of this original work, Gabriel Hernandez, that the name be altered to reflect the true power of the method and not on the construct on which it is loosely based.

- *Why the utility-based compromise Decision Support Problem?* Quite simply, the u-cDSP was chosen for this work because it provides a construct for solving multi-objective, non-linear design problems. The u-cDSP is a decision support construct that permits mathematically rigorous modeling of designer preferences. Its use also permits the designer to accept "satisficing" solutions instead of the optimal solution, which may be impossible to find due to practical reasons. The u-cDSP also provides a standard template with which to structure a decision. Finally, the u-cDSP has proven to be valid in the development of product platforms in the literature (Seepersad, 2001; Seepersad et al., 2002). While there are limitations associated with its use, the u-cDSP provides a rigorous method for handling multiple objectives.

Regarding the second question frequently asked of design methodology research, the role of the designer in the application of the PPCTM is very important. The PPCTM is in no way a replacement for a team of designers. Designers and engineers are needed to develop the product, develop the simulation models and associated code, as well as develop the modes of managing variety concepts and implementation techniques. This method cannot be used without the close supervision and knowledge of a designer. The results of this method only inform a team of designers of the most ideal manner of

structuring the ranges of commonalization techniques for the product family. It is meant to shorten the number of design iterations as well as improve upon the objectives established by the designers.

Following the answer of these two frequently asked questions, I present some open questions for my benefit. *Is the result of the use of the PPCTM worth the computational effort? What is the benefit of its use?* Although the results presented in Chapters 4 and 5 show a direct benefit to the use of the PPCTM, these results do not explicitly show the *value* of its use. In fact, upon closer inspection, the results of Chapters 4 and 5 only explicitly show that some form of commonalization is better for the platform than the two extremes. This only proves that the concepts of mass-customization are valid. Although the method is extremely computationally expensive, I firmly believe that the use of the PPCTM is the best manner currently available to model the complexity of product/process platform design. I do not doubt however, that there is a more efficient manner in which to solve the problem. The only manner in which to fully evaluate the usefulness of the PPCTM is through its use in a design that is carried out from conceptualization to finish.

Another question brought to my attention frequently is, *Should research be continued towards this direction?* As can be seen in the list of future work possibilities, there is definitely room for the improvement of the augmented PPCTM. The question remains, however, what is left in product platform design methods? I have recently met with several engineers working with Philips who informed me that they felt that product platform design only looked good on paper – it did not prove to be of any advantage in application. While some may take their stance as an invitation to look elsewhere for

research directions, I instead suggest that it be used as fuel on which future research is kindled. A current limitation of product platform design techniques is their limited scope. If instead of simply attempting to design techniques for handling commonality around a single product, the research focus should be shifted towards the commonality of the supply chain. As corporations become more global in their design, manufacture, and distribution, the connections between these divisions provide opportunities for improvement via product platforms. Another potential avenue for future work could be in the application of PPCTM concepts along the supply chain.

Personally, I am interested in connection between design and manufacturing in the paradigm of mass customization. The application of the augmented PPCTM to the realm of production process design provides evidence of this passion. My current interest lies in the development of a manufacturing technology capable of providing a boost in the paradigm of mass customization and in the way products are designed. As a result, this is the direction of my Ph.D. research.

Before moving onto this domain of research, I take time to reflect on my accomplishments from completing this thesis. I am excited about the direction of the PPCTM. It is the only method that synthesizes multiple modes of managing product customization. It is the only product platform design method that is able to explore and evaluate the possibility of non-uniform demand. It is the only method that relates levels of product demand to the resulting product platform. My augmentations have established a new construct in the realm of process design. I am excited to see how valuable the development of a "process platform" will be.

Besides the goals listed at the beginning of Section 6.4, my main hope for this work is that I have added strength and validity to Hernandez's Product Platform Constructal Theory Method. I believe that is an exciting and novel manner in which to visualize product and process platform problems. I hope that this work will assist in its introduction and acceptance into the design community. Furthermore, it is my hope that in through the completion of this thesis I have developed a few interesting ideas and questions with regards to both product platform design and process platform design.

APPENDIX A

JAVA CODE FOR THE PRESSURE VESSEL EXAMPLE PROBLEM

In this appendix, the computer code for the implementation of the pressure vessel example problem (see Chapter 4) is presented. This computer code is programmed in the JAVA computer language. A brief description of each class file and the functions of each file are presented in Appendix A.1. The actual code is presented in Appendix A.2.

A.1 LIST OF CLASS FILES AND FUNCTIONS

A.1.1 List of Class Files

clsArrayOutput

The instantiation of this class creates a BufferedWriter, which is used to write to an output file. The resulting output file contains detailed information of the properties of each node of the space of customization.

clsDemandOutput

Similar to the previous class, this class creates a BufferedReader to create an output file. This output file contains the amount of demand for each node of the market space. This output file is used to create the graphs of each demand scenario.

clsMarketSpace

This class provides a function for the discretization of a two-dimensional geometric space. Methods are also provided for the instantiation of the data to be stored for each

node in the space. Mathematical calculations for modeling several different types of non-uniform demand are also provided.

clsPressureVessel

This class is the main class of the computer code and provides methods for the program's instantiation from the DOS prompt. It serves as the central hub in which the solution algorithm is presented and the other classes are called.

clsResultOutput

This class is used to create an output file that captures the history of the product platforms that, at their time of investigation, provide the highest utility.

A.1.2 List of Functions

clsArrayOutput

addText: adds the pertinent data of each node of the customization space to the output file

addProfit: adds the resulting average profit of the current iteration in the analysis to the output file.

addResults: adds the final, best results to the end of the file.

close: closes the output file

clsMarketSpace

getNodes: calculates and establishes the number of nodes needed to discretize the space

fillDArray: models the demand of the market as a set of concentric squares with discrete levels of demand

fillDArray2: models the demand of the market as a linear continuous function of volume and pressure

fillDArray3: models the demand of the market as a set of discrete values along the volume dimension

fillDArray4: models the demand of the market as a two-dimensional Gaussian distribution

fillDArray5: models the demand of the market as a random distribution

fillPrice: models the selling price of vessels of the market as a function of volume and pressure

getTotDemand: sums the demand of all product variants across the market space

getNodeArray: returns the array of nodes as created by the getNodes function

getXDim: returns the volume dimension (number of nodes) of the space of customization

getYDim: returns the pressure dimension of the space (number of nodes) of customization

getNodeDim: returns the total number of nodes in the discretized space of customization

initCost: sets the cost of each node to 0 for the next iteration

initVmax: sets the associated maximum volume of each node to 0 for the next iteration

initPmax: sets the associated maximum pressure of each node to 0 for the next iteration

initRmin: sets the associated radius of each node to 0 for the next iteration

initVflag: initializes the flag raised when the volume of the node has been established

initPflag: initializes the flag raised when pressure volume of the node has been established

initRflag: initializes the flag raised when the radius of the node has been established

clsPressureVessel

factorCheck: This function divides a number into another and checks to see if indeed one number is a multiple of the other

Cost: This use of this function provides a designer the ability to calculate the cost of a single pressure vessel product variant

Rmin: By inputting a specific volume, this function will identify the minimum feasible radius (using the largest feasible length). The radius is found by applying the bisection method root-finding technique.

clsResultOutput

addText: similar to clsArrayOutput, this function adds text to the appropriate output file

close: similar to clsArrayOutput, this function closes the appropriate output file

A.2 SOFTWARE CODE

clsArrayOutput

```
/*
 * clsArrayOutput.java
 */

 /* -- NOTES --
  * This class creates a BufferedWriter
  * it adds text with the addText() function
  * which requires an input of the nodeArray
  * it outputs a text file to be analyzed with Excel
 */

 import java.io.*;

 public class clsArrayOutput
 {
      BufferedWriter out;
      //---- Creates new clsArrayOutput ----//
      public clsArrayOutput(double res)
      {
            try
            {
                  out = new BufferedWriter(new FileWriter("NodeInfo_" +
                        res + ".txt"));
                  out.write("Resolution = " + res);
                  out.write("\n");
            }
            catch (IOException e)
            {
                  System.err.println("FileStream error " + e);
            }
```

```
}//end of clsArrayOutput constructor


//----FUNCTIONS----//

//FUNCTION: addText//
//function adds text to the opened file
public void addText(double dV3, double dP2, double dV1, double
         dP1, double[][] nodeArray, int arraylength)
{
      try
      {

out.write("+++++++++++++++++++++++++++++++++++++++++++++");
            out.write("\n");
            Double ddV3 = new Double(dV3);
            String sdV3 = ddV3.toString();
            out.write("dV3: ");
            out.write(sdV3);
            out.write("   ");
            Double ddP2 = new Double(dP2);
            String sdP2 = ddP2.toString();
            out.write("dP2: ");
            out.write(sdP2);
            out.write("   ");
            Double ddV1 = new Double(dV1);
            String sdV1 = ddV1.toString();
            out.write("dV1: ");
            out.write(sdV1);
            out.write("   ");
            Double ddP1 = new Double(dP1);
            String sdP1 = ddP1.toString();
            out.write("dP1:  ");
            out.write(sdP1);
            out.write("\n");

            out.write("Node#   XLoc   Yloc   Demand   Cost
                      MaxV   MaxP   Rmin   Price");
            out.write("\n");

            for (int i=0; i<arraylength; i++)
            {
                  Integer k = new Integer(i);
                  String r = k.toString();
                  out.write(r);
                  out.write("        ");

                  for (int j= 0; j<8; j++)
                  {
                        Double d = new Double(nodeArray[i][j]);
                        String s = d.toString();
                        out.write(s);
                        out.write("     ");
                  }
                  out.write("\n");
            }
      }
```

```
        catch (IOException e)
        {
                System.err.println("FileStream error " + e);
        }
}//end addText()

//FUNCTION: addProfit()
//function adds avg. profit to file
public void addProfit(double profit)
{
        try
        {
                Double d = new Double(profit);
                String s = d.toString();
                out.write("Avg. Profit = $" + s);
                out.write("\n");
        }
        catch (IOException e)
        {
                System.err.println("FileStream error " + e);
        }
}

//FUNCTION: addResults()
//function adds final results to end of file
public void addResults(double dV3, double dP2, double dV1, double
            dP1, double Profit)
{
        try
        {
                out.write("\n");
                out.write("---- ANALYSIS FINISHED ----" + "\n");
                out.write("dV3 = " + dV3 + "\n");
                out.write("dP2 = " +    dP2 + "\n");
                out.write("dV1 = " + dV1 + "\n");
                out.write("dP1 = " + dP1 + "\n");
                out.write("Profit = " + Profit + "\n");
        }
        catch (IOException e)
        {
                System.err.println("FileStream error " + e);
        }
}

//FUNCTION: close//
//function closes file
public void close()
{
        try
        {
                out.close();
        }
        catch (IOException e)
        {
                System.err.println("FileStream error " + e);
        }
}//end close()
```

```
}//end of clsArrayOutput
```

clsDemandOutput

```
/*
 * clsDemandOutput.java
 */

 /* -- NOTES --
  * This class creates a BufferedWriter
  * it adds text with the addText() function
  * which requires an input of the nodeArray
  * it outputs a text file to be analyzed with Excel
 */

 import java.io.*;

 public class clsDemandOutput
 {
      BufferedWriter out;
      //---- Creates new clsDemandOutput ----//
       public clsDemandOutput(double res, double[][] nodearray, int
                              xdim, int ydim)
      {
            try
            {
                  out = new BufferedWriter(new FileWriter("Demand_" +
                                          res + ".txt"));
                  out.write("Resolution = " + res);
                  out.write("\n");

                  for (int i=0; i<xdim; i++)
                  {

                        for (int j=0; j<ydim; j++)
                        {
                              out.write(" " + nodearray[ j + i*xdim][2]
                                          + " ");
                        }
                        out.write("\n");
                  }


                  out.close();
            }
            catch (IOException e)
            {
                  System.err.println("FileStream error " + e);
            }
      }//end of clsDemandOutput constructor

 }//end of clsDemandOutput
```

clsMarketSpace

```
/*
 * clsMarketSpace.java
 */

 /* -- NOTES --
  * This class requires an input of
  * the dimensions of a 2D market space
  * and a resolution number.
  * The output is an array of nodes.
 */

 import java.lang.Number.*;

 public class clsMarketSpace
 {

      //---- Constants ----//
      double res; //resolution variable
      double xo; //initial x value
      double dimx; //length of x axis (continuous)
      double yo; //initial y value
      double dimy; //length of y axis (continuous)

      int xint = 0; //dimension of x-axis array
      int yint = 0; //dimension of y-axis array

      double nodearray[][]; //a 2D array that stores all info about a
                              node

      //---- Creates new clsMarketSpace ----//
       public clsMarketSpace(double dres, double dxo, double ddimx,
                              double dyo, double ddimy)
      {
            //---- Problem Dependent Variables ----//

            res = dres;
            xo = dxo;
            dimx = ddimx;
            yo = dyo;
            dimy = ddimy;

            //---- Analysis ----//

            //make array of dim(xint,yint), start with xo, add res, end
                  with xn
            getNodes();  //establishes number of nodes
            //create nodearray[]
            //# of rows = # of nodes (# of X x # of Y)
            //# of columns = 11 (xlocation, ylocation, demand, cost,
                        maxV, maxP, Rmin, Price, flagV, flagP, flagR)
            nodearray = new double[xint*yint][11];
            nodearray[0][0] = xo;
            nodearray[0][1] = yo;
```

```
            //fill nodearray with x and y locations
            for (int k=1; k<(xint*yint); k++)
            {
                  for (int i=1; i<xint; i++)
                  {
                        nodearray[k][0] = nodearray[k-1][0] + res;
                        nodearray[k][1] = nodearray[k-1][1];
                        k++;
                  }
                  if (k<(xint*yint))
                  {
                        nodearray[k][0] = xo;
                        nodearray[k][1] = nodearray[k-1][1] + res;
                  }
            }
            //just in case the res is not a factor of the dim...
            nodearray[xint-1][0] = xo + dimx;
            nodearray[yint-1][0] = yo + dimx;

            //fill in demand scenario
             fillDArray(); //Concentric square scenario; Demand
                           Scenarios 1-3
//          fillDArray2(); //linear scenario; Demand Scenario 4 & 5
//          fillDArray3(); //discrete scneario; Demand Scenario 6 & 7
//          fillDArray4(); //Gaussian distribution (normal) scenario;
            Demand Scenario 8 - 11
//          fillDArray5(); //random demand; Demand Scenario 12 & 13

            fillPrice(); //fill in selling price scenario

            clsDemandOutput DemandOutput = new clsDemandOutput(res,
                                           nodearray, xint, yint);


      } //end constructor: clsMarketSpace()


      //------- FUNCTIONS -------//

      //---- FUNCTION: getNodes() ----//
      public void getNodes()
      {
            //used in the constructor to establish the number of nodes
            Double x = new Double (dimx / res);
            Double y = new Double (dimy / res);

            xint = x.intValue() + 1;
            yint = y.intValue() + 1;

            double xd = xint+0.0;
            double yd = yint+0.0;
            double decimaltestx = (dimx/res) - xd;
            double decimaltesty = (dimy/res) - yd;
            if (decimaltestx > 0)
            {
                  xint = xint++;
```

```
                System.out.println("ERROR: Resolution is not a factor
                                    of Dimension!");
            }
            if (decimaltesty > 0)
            {
                yint = yint++;
                System.out.println("ERROR: Resolution is not a factor
                                    of Dimension!");
            }
        }//end getArrayDim

        //---- FUNCTION: fillDArray() ----//
        public void fillDArray()
        {
            //used in constructor to
            //fill nodearray[][2] with appropriate demand scenario

            //define the demand scenario
            // for this scenario, demand is modeled as a set of
            // concentric squares (D1 on outer, D2 in middle, and D3 in
               inner)


            //Demand Scenario 1
            double D1 = 150.0; //demand level 1 of 3
            double D2 = 250.0; //demand level 2 of 3
            double D3 = 400.0; //demand level 3 of 3

/*
            //Demand Scenario 2
            //uniform demand of 800 total
            double D1 = 50.0; //demand level 1 of 3
            double D2 = 50.0; //demand level 2 of 3
            double D3 = 50.0; //demand level 3 of 3
*/

/*
            //Demand Scenario 3
            //crazy demand #1
            double D1 = 0.0; //demand level 1 of 3
            double D2 = 100.0; //demand level 2 of 3
            double D3 = 300.0; //demand level 3 of 3
*/

            double x1 = 14; //1st cut-off for D1 on x-axis
            double x2 = 18;
            double x3 = 22;
            double x4 = 26;
            double y1 = 14; //1st cut-off for D2 on y-axis
            double y2 = 18;
            double y3 = 22;
            double y4 = 26;

            for (int k=0; k<(xint*yint); k++)
            {
                if (nodearray[k][0] < (xo + dimx + 1.0)  &&
                nodearray[k][1] < (yo + dimy + 1.0))
```

```
                {
                        nodearray[k][2] = D1;
                }
                if (nodearray[k][0] < x4 && nodearray[k][1] < y4 &&
                nodearray[k][0] > x1 && nodearray[k][1] > y1)
                {
                        nodearray[k][2] = D2;
                }
                if (nodearray[k][0] < x3 && nodearray[k][1] < y3 &&
                nodearray[k][0] > x2 && nodearray[k][1] > y2)
                {
                        nodearray[k][2] = D3;
                }
        }
} //end of fillDArray


//---- FUNCTION: fillDArray2() ----//
public void fillDArray2()
{
        //used in constructor to
        //fill nodearray[][2] with appropriate demand scenario

        //define the demand scenario
        // for this scenario, demand is modeled as a linear
                function of V and P
        // D = X*V + Y*P
/*
        //Demand Scenario 4
        double x = 100;
        double y = 100;
*/
        //Demand Scenario 5
        double x = 10;
        double y = 10;


        for (int k=0; k<(xint*yint); k++)
        {
                nodearray[k][2] = (x * (nodearray[k][0] / 10) ) + (y
                * (nodearray[k][1] / 10) );
        }
} //end of fillDArray2()


//---- FUNCTION: fillDArray3() ----//
public void fillDArray3()
{
        //used in constructor to
        //fill nodearray[][2] with appropriate demand scenario

        //define the demand scenario
        // for this scenario, demand is modeled as a set of
        discrete values depending on V

/*
        //Demand Scenario 6
```

```
            double x1 = 14; //1st cut-off for D1 on x-axis
            double x2 = 18;
            double x3 = 22;
            double x4 = 26;

            double D1 = 400;
            double D2 = 300;
            double D3 = 200;
            double D4 = 100;
            double D5 = 50;
*/

            //Demand Scenario 7
            double x1 = 14; //1st cut-off for D1 on x-axis
            double x2 = 18;
            double x3 = 22;
            double x4 = 26;

            double D1 = 10;
            double D2 = 10;
            double D3 = 10;
            double D4 = 10;
            double D5 = 10;


            for (int k=0; k<(xint*yint); k++)
            {
                  nodearray[k][2] = D5;

                  if (nodearray[k][0] < x4  )
                  {
                        nodearray[k][2] = D4;
                  }
                  if (nodearray[k][0] < x3 )
                  {
                        nodearray[k][2] = D3;
                  }
                  if (nodearray[k][0] < x2 )
                  {
                        nodearray[k][2] = D2;
                  }
                  if (nodearray[k][0] < x1 )
                  {
                        nodearray[k][2] = D1;
                  }
            }
      } //end of fillDArray3()


      //---- FUNCTION: fillDArray4() ----//
      public void fillDArray4()
      {
            //used in constructor to
            //fill nodearray[][2] with appropriate demand scenario

            //define the demand scenario
```

```
            // for this scenario, demand is modeled as a 3D Gaussian
            (normal) distribution

/*
            //Demand Scenario 8
            double mean = 20.0;
            double xvariance = 15.0;
            double yvariance = 15.0;
            double A = 200.0;
*/

/*
            //Demand Scenario 9
            double mean = 12.0;
            double xvariance = 15.0;
            double yvariance = 5.0;
            double A = 200.0;
*/

/*
            //Demand Scenario 10
            double mean = 24.0;
            double xvariance = 15.0;
            double yvariance = 5.0;
            double A = 200.0;
*/

            //Demand Scenario 11
            double mean = 20.0;
            double xvariance = 5.0;
            double yvariance = 5.0;
            double A = 200.0;

            double xstddev = java.lang.Math.sqrt(xvariance); //standard
            deviation, x
            double ystddev = java.lang.Math.sqrt(yvariance); //standard
            deviation, y

            double xD;
            double yD;

            double D;
            double X;
            double Y;

            for (int k=0; k<(xint*yint); k++)
            {
                  X = nodearray[k][0];
                  Y = nodearray[k][1];

                  xD = A * (1 / (xstddev *
                       java.lang.Math.sqrt(2*java.lang.Math.PI))) *
                       java.lang.Math.exp(-0.5 * java.lang.Math.pow(((X
                       - mean)/xstddev),2));
                  yD = A * (1 / (ystddev *
                       java.lang.Math.sqrt(2*java.lang.Math.PI))) *
```

```
                    java.lang.Math.exp(-0.5 * java.lang.Math.pow(((Y
                    - mean)/ystddev),2));

                D = xD * yD;

                nodearray[k][2] = D;
            }

        } //end of fillDArray4()

        //---- FUNCTION: fillDArray5() ----//
        public void fillDArray5()
        {

            //Demand Scenario 12
//          double maxDemand = 500.0;

            //Demand Scenario 13
            double maxDemand = 10.0;

            double d;
            java.util.Random ranum = new java.util.Random();
            double demand;

            for (int k=0; k<(xint*yint); k++)
            {
                d = ranum.nextDouble(); //generates a random double
                    from 0.0 to 1.0 (2^53 possible float values -
                    check 1.4.1 API)
                demand = d * maxDemand;
                nodearray[k][2] = demand;
            }

        } //end of fillDArray5()

        //---- FUNCTION: fillPrice() ----//
        public void fillPrice()
        {
            //establish a selling price of the pressure vessels
            //based on V and P
            //Price = $x*V + $x*P
            for (int i=0; i<(xint*yint); i++)
            {
                nodearray[i][7] = 185*nodearray[i][0] +
                                  185*nodearray[i][1];
            }
        } //end of fillPrice()


        //---- FUNCTION: getTotDemand() ----//
        public double getTotDemand()
        {
            double totalDemand = 0.0;
            for (int i = 0; i<(xint*yint); i++)
            {
                totalDemand = totalDemand + nodearray[i][2];
            }
```

```
        return totalDemand;
} //end of getTotDemand()


//---- FUNCTION: getNodeArray() ----//
public double[][] getNodeArray()
{
        return nodearray;
} //end of getNodeArray


//---- FUNCTION: getXDim() ----//
public int getXDim()
{
        return xint;
}//end getXDim


//---- FUNCTION: getYDim() ----//
public int getYDim()
{
        return yint;
} //end of getYDim


//---- FUNCTION: getNodeDim() ----//
public int getNodeDim()
{
        return xint*yint;
} //end of getNodeDim()


//---- FUNCTION: initCost() ----//
public void initCost()
{
        //initializes the cost column in the array
        for (int i=0; i<(xint*yint); i++)
        {
                nodearray[i][3] = 0.0;
        }
}//end of initCost()


//---- FUNCTION: initVmax() ----//
public void initVmax()
{
    //initializes the maxV column in the array
        for (int i=0; i<(xint*yint); i++)
        {
                nodearray[i][4] = 0.0;
        }
}//end of initVmax()


//---- FUNCTION: initPmax() ----//
public void initPmax()
{
```

```
   //initializes the maxP column in the array
      for (int i=0; i<(xint*yint); i++)
      {
            nodearray[i][5] = 0.0;
      }
}//end of initPmax()


//---- FUNCTION: initRmin() ----//
public void initRmin()
{
   //initializes the Rmin column in the array
      for (int i=0; i<(xint*yint); i++)
      {
            nodearray[i][6] = 0.0;
      }
}//end of initRmin()


//---- FUNCTION: initR() ----//
public void initR()
{
   //initializes the R column in the array
      for (int i=0; i<(xint*yint); i++)
      {
            nodearray[i][7] = 0.0;
      }     }//end of initR()


//---- FUNCTION: initVflag() ----//
public void initVflag()
{
   //initializes the Vflag column in the array
      for (int i=0; i<(xint*yint); i++)
      {
            nodearray[i][8] = 0.0;
      }
}//end of initVflag()


//---- FUNCTION: initPflag() ----//
public void initPflag()
{
   //initializes the Pflag column in the array
      for (int i=0; i<(xint*yint); i++)
      {
            nodearray[i][9] = 0.0;
      }
}//end of initPflag()


//---- FUNCTION: initRflag() ----//
public void initRflag()
{
   //initializes the Rflag column in the array
      for (int i=0; i<(xint*yint); i++)
      {
```

```
                nodearray[i][10] = 0.0;
            }
        }//end of initRflag()


    //end of FUNCTIONS


    }//end clsMarketSpace
```

clsPressureVessel

```
/*
 * clsPressureVessel.java
 */

public class clsPressureVessel {

        public static void main (String[] args)
        {
                new clsPressureVessel();
        }

        //---- Problem Constraints ----//
        double Tl = 6.35; //lower thickness limit, mm
        double Tu = 76.2; //upper thickness limit, mm (multiples of 6.35)
        double Lu = 7.0; //upper length limit, m
        double Ru = 4.0; //upper radius length, m
        double pi = java.lang.Math.PI; //pi, 3.14...

        double dP1best = 0.0;
        double dP2best = 0.0;
        double dV1best = 0.0;
        double dV3best = 0.0;
        double Profitbest = 0.0;

        //---- Creates new clsPressureVessel ----//
        public clsPressureVessel()
        {
                //---- First, define and discretize space ----//
                double res = 0.5; //resolution of discretization

                double vo = 10.0; //initial point of x-dimension of space
                double dimv = 20; //length of x axis
                double po = 10.0; //initial point of y-dimension of space
                double dimp = 20; //length of y axis

                //Use MarketSpace class to define Market
                //get the node array ... a place to store each node's info
                clsMarketSpace MarketSpace = new clsMarketSpace(res, vo,
                                                dimv, po, dimp);
                double nodeArray[][] = MarketSpace.getNodeArray();

                //ues to find the lowest profit combo out of all iterations
                double averageprofit = 0.0;
```

```
//establish output file
clsArrayOutput ArrayOutput = new clsArrayOutput(res);
clsResultOutput ResultOutput = new clsResultOutput(res);

//ALGORITHM DESCRIPTION
//iterate through dV2 (=dV1)
//assign each node its Vmax (will be used for Lo)
//from Vmax, find min feasible radius
//assume this radius is the cheapest option
//make it constant across a dP2
//iterate through dP2 (must be a multiple of Pdim)
//iterate dP1 (must be a multiple of dP2)
//assign each node its Pmax (each node has its own V)
//iterate through dV3 (a cutoff point for combination)
//calculate profit of each node (needs R, Pmax, V, Lo)
//be sure to take care in evaluation of CEquip
//store largest profit and report at end
//also print out file with info on each node!

double dV1; //size of first-stage volume element
double V2; //largest V of the current space

//iterate through dV1
for (int j=1; j<((MarketSpace.getXDim()/2)+1); j++)
{
      V2 = nodeArray[j][0];
      dV1 = V2 - nodeArray[0][0];
      int dVnode = j; //number of nodes in dV1

      //check to see if dV is a multiple of the dimension
          of the space
      int Vflag = factorCheck(dimv, dV1);
      if (Vflag > 0)
      {;}
      else //if it is a multiple keep on truckin'
      {
      //number of elements in V direction
      int Velement = (MarketSpace.getXDim()-1)/dVnode;

      //loop through elements and assign each node its maxV
      double Vmax;
      double Rmin;
      for (int z=1; z<Velement+1; z++)
      {
      Vmax = nodeArray[z*dVnode][0];
      Rmin = Rmin(Vmax);
      //fill previous nodes with Vmax if they haven't been
             filled yet
      for (int y=0; y<((z*dVnode)+1); y++)
      {
      if(nodeArray[y][8]<1.0)
      {
      nodeArray[y][4] = Vmax;
      nodeArray[y][6] = Rmin;
      nodeArray[y][8] = 1.0;
      }
      else {;}
```

```
//fill all nodes above 1st line (they have same Vmax)
for(int x=1; x<MarketSpace.getYDim(); x++)
{
if(nodeArray[y+x*MarketSpace.getXDim()][8]<1.0)
{
nodeArray[y+x*MarketSpace.getXDim()][4] = Vmax;
nodeArray[y+x*MarketSpace.getXDim()][6] = Rmin;
nodeArray[y+x*MarketSpace.getXDim()][8] = 1.0;
}
else{;}
}
}
}//end of Vmax loop
double dP2; //size of second-stage pressure element
double P3; //largest P of current space

//iterate through dP2
for (int k=1; k<((MarketSpace.getYDim()/2)+1) ; k++)
{
P3 = nodeArray[k*MarketSpace.getXDim()][1];
dP2 = P3 - nodeArray[0][1];

//check to see if dP2 is a multiple of the dimension
    of the space
int dP2flag = factorCheck(dimp, dP2);
if (dP2flag > 0)
{;}
else //if it is a multiple keep on truckin'
{
int dP2node = k; //number of node in dP2
//number of elements in P direction
int P2element = (MarketSpace.getYDim()-1)/dP2node;

double dP1; //size of first-stage pressure element
double P2;  //largest P of the current space

//iterate through dP1
for (int i=1; i < (dP2node+1); i++)
{
P2 = nodeArray[i*MarketSpace.getXDim()][1];
dP1 = P2 - nodeArray[0][1];

//check to see if dP1 is a multiple of dP2
int dP1flag = factorCheck((P3-po), dP1);
if (dP1flag > 0)
{;}
else //if it is a multiple keep on truckin'
{
int dP1node = i; //number of nodes in dP1
//number of elements in P direction
int P1element = (MarketSpace.getYDim()-1)/dP1node;
//loop through elements and assign each node its maxP
nodeArray[0][5] = P2;
nodeArray[0][9] = 1.0; //set Pflag to 1
double Pmax;
for (int a=0; a<P1element; a++)
{
```

```
                    int PmaxNode = MarketSpace.getXDim()*dP1node +
                          a*dP1node*MarketSpace.getXDim();
                    Pmax = nodeArray[PmaxNode][1];
                    nodeArray[PmaxNode][5] = Pmax;
                    nodeArray[PmaxNode][9] = 1.0;
                    //fill nodes below Pmax line and check Pflag!
                    int d = PmaxNode;
                    while (d > 0)
                    {
                    d = d - MarketSpace.getXDim();
                    if (nodeArray[d][9] < 1.0)
                    {
                    nodeArray[d][5] = Pmax;
                    nodeArray[d][9] = 1.0;
                    }
                    else {;}
                    }
                    }//end of Pmax loop
                    //fill nodes on constP line with Pmax
                    int b = 0;
                    while (b < (MarketSpace.getNodeDim() -
                          MarketSpace.getXDim())+1)
                    {
                    for(int e=1; e<MarketSpace.getXDim() ; e++)
                    {
                    nodeArray[b+e][5] = nodeArray[b][5];
                    nodeArray[b+e][9] = 1.0;
                    }
                    b = b + MarketSpace.getXDim();
                    }

//ITERATE THROUGH POSSIBLE dV3s
//(not restricted to being a multiple of Vdim - looking for cutoff
point)
//FIRST, TRY A POINT WHERE dV3 = VDim
//ESTABLISH A CUTOFF POINT (starting at half of XDim - only combine 2
vessels),
//RECALCULATE COST ASSOCIATED WITH COMBINING VESSELS FOR THOSE ABOVE
THAT POINT
//CALC AVERAGE PROFIT... FIND BEST dV3 FOR EACH dP2/dV1/dP1 COMBO
                    double dV3; //size of third space element
                    double V3; //largest V of the space... aka "cutoff
                              point"
                    //first try when there is no dV3 - no cutoff
                    dV3 = dimv;
                    //calculate cost for each node
                    for (int n=0; n<MarketSpace.getNodeDim(); n++)
                    {
                    double V = nodeArray[n][0];
                    double R = nodeArray[n][6];
                    double maxV = nodeArray[n][4];
                    double maxP = nodeArray[n][5];
                    double Lo = (maxV / (pi*R*R)) - ((4.0/3.0)*R);
                    double L = (V / (pi*R*R)) - ((4.0/3.0)*R);
                    double cost = Cost(maxP, V, R, L, Lo);
                    nodeArray[n][3] = cost;
                    }
```

```
//find avg profit of entire space
double newprofit = 0;
double totalDemand = 0;
for (int q = 0; q<MarketSpace.getNodeDim(); q++)
{
//demand * (selling price - cost)
newprofit = newprofit +
 nodeArray[q][2]*(nodeArray[q][7] - nodeArray[q][3]));
totalDemand = nodeArray[q][2] + totalDemand;
}
// CALCULATE PRESS COST with R of each dV1
double CEquip = 0;
for (int p=1; p<(Velement+1); p++)
{
double rad = nodeArray[p*dVnode][6];
CEquip = CEquip + (500000 + (50000*rad));
}

//include cost of different Lo's
double Locost = 250*Velement;
double newprofit1 = newprofit - CEquip - Locost;

//divide by total demand of space to take average
newprofit1 = newprofit1 / totalDemand;

//is it less than previous avgprofit for this
    iteration?
//if so store it in the array
if (newprofit1 > averageprofit)
{
averageprofit = newprofit1;
dP1best = dP1;
dP2best = dP2;
dV1best = dV1;
dV3best = dimv;
Profitbest = averageprofit;
//write results to file
ArrayOutput.addText(dV3, dP2, dV1, dP1, nodeArray,
     MarketSpace.getNodeDim());
ArrayOutput.addProfit(newprofit1);
}

 ResultOutput.addText(dV3, dP2, dV1, dP1, newprofit,
      CEquip, totalDemand, newprofit1);

//now iterate through dV3s
for (int c=((MarketSpace.getXDim()/2)+1);
      c<MarketSpace.getXDim(); c++)
{
int dV3node = c;
V3 = nodeArray[dV3node-1][0];
dV3 = V3 - nodeArray[0][0];

//check to see if dV3 is a multiple of dV1
int dV3flag = factorCheck(dV3, dV1);
if (dV3flag > 0)
{;}
```

```
else //if it is a multiple keep on truckin'
{

//calculate cost for each node
//need R, Pmax, Lo, and V(node)
for (int n=0; n<MarketSpace.getNodeDim(); n++)
{
double V = nodeArray[n][0];
double R = nodeArray[n][6];
double maxV = nodeArray[n][4];
double maxP = nodeArray[n][5];
double Lo = (maxV / (pi*R*R)) - ((4.0/3.0)*R);

double L = (V / (pi*R*R)) - ((4.0/3.0)*R);

double cost = Cost(maxP, V, R, L, Lo);

nodeArray[n][3] = cost;
}

//evaluate the cost of all nodes past V3 as 2xhalf-
   size
for (int d = dV3node; d<MarketSpace.getXDim(); d++)
{
for (int e=0; e<MarketSpace.getYDim(); e++)
{
int node = d/2 + (e*MarketSpace.getXDim());
      //location of half of current node
nodeArray[(d+(e*MarketSpace.getXDim()))][3] = 2 *
          nodeArray[node][3];
}
}

//find avg profit of entire space
newprofit = 0;
totalDemand = 0;
for (int q = 0; q<MarketSpace.getNodeDim(); q++)
{
//demand * (selling price - cost)
newprofit = newprofit +
     (nodeArray[q][2]*(nodeArray[q][7] -
     nodeArray[q][3]));
totalDemand = nodeArray[q][2] + totalDemand;
}


// CALCULATE PRESS COST with R of each dV1 (based on
   dV3)
CEquip = 0;
int o = 1;
int onode = dVnode;
while (onode < dV3node)
{
double rad = nodeArray[onode][6];
CEquip = CEquip + (500000 + (50000*rad));
o++;
onode = o*dVnode;
```

```
        }

        newprofit1 = newprofit - CEquip - Locost;

        //divide by total demand of space to take average
        newprofit1 = newprofit1 / totalDemand;

        //is it less than previous avgprofit for this
                iteration?
        //if so store it in the array
        if (newprofit1 > averageprofit)
        {
        averageprofit = newprofit1;
        dP1best = dP1;
        dP2best = dP2;
        dV1best = dV1;
        dV3best = dV3;
        Profitbest = averageprofit;
        //write results to file
        ArrayOutput.addText(dV3, dP2, dV1, dP1, nodeArray,
                    MarketSpace.getNodeDim());
        ArrayOutput.addProfit(newprofit1);
        }

        ResultOutput.addText(dV3, dP2, dV1, dP1, newprofit,
              CEquip, totalDemand, newprofit1);

        MarketSpace.initCost();
        }//end of else in dV3 iteration

        }//end of dV3 iteration

        //initialize flags for next iteration
        MarketSpace.initPmax();
        MarketSpace.initPflag();
    }//end of else of dP1
    }//end of dP1

    }//end of else of dP2
    } //end of dP2

    MarketSpace.initVmax();
    MarketSpace.initVflag();
    } //end of else of dV1
    }//end of dV1

    //write final results to end of file
    ArrayOutput.addResults(dV3best, dP2best, dV1best, dP1best,
          Profitbest);

    //close output files
    ArrayOutput.close();
    ResultOutput.close();

    //write final results to system
    System.out.println("---- ANALYSIS FINISHED ----");
    System.out.println("dV3 = " + dV3best);
```

```
        System.out.println("dP2 = " + dP2best);
        System.out.println("dV1 = " + dV1best);
        System.out.println("dP1 = " + dP1best);
        System.out.println("Profit = " + Profitbest);

    } //end of clsPressureVessel() constructor

    //---- FUNCTIONS ----//

    //FUNCTION: factorCheck()//
    public int factorCheck(double n, double m)
    {
        Double x = new Double(n/m);
        int y = x.intValue();
        double z = y+0.0;
        double fractioncheck = (n/m) - z;
        //System.out.println("fractioncheck = " + fractioncheck);
        int flag = 0;
        if (fractioncheck < 0 || fractioncheck > 0)
        {
            flag = 1;
        }
        return flag;
    }//end of factorCheck()

    //FUNCTION: Cost()//
    public double Cost(double xP, double xV, double xR, double xL,
                double xLo)
    {
        double P = xP; //Pressure, MPa
        double V = xV; //Volume, m^3
        double R = xR; //Radius, m
        double L = xL; //Length, m
        double Lo = xLo; //Raw Length, m

        //---- Constants ----//
        double sigy = 1077; //Yield strength, MPa
        double dens = 7800; //density, kg/m^3
        double Cs = 0.8; //processed shell stell cost, $/kg
        double Ch = 2.0; //cost of forged steel for head, $/kg
        double Cp = 0.3; //cost of raw steel plate, $/kg
        double Cw = 15; //cost of welding material, $/kg

        double Ts = (R*1000)* (P / (sigy - (0.6*P))); //shell
            thickness, mm
        double Th = (R*1000)* (P / ( (2*sigy) - (0.2*P) )); //head
            thickness, mm

         double CWaste = 2 * pi * dens * Cp * (Ts/1000) * R * (Lo -
            L); //cost of wasted material, $
        double CMat = 2 * pi * dens * ( (Cs*R*(Ts/1000)*L) +
            (Ch*R*R*(Th/1000)) ) + CWaste; //material cost, $
        double VLWeld = (4.0/9.0) * pi * (Ts/1000) * (Ts/1000) * L;
            //volume of long weld, m^3
        double CLongWeld = VLWeld * dens * Cw; //cost of long weld
        double VCWeld = (8.0/9.0) * pi * pi * (Ts/1000) * (Ts/1000)
            * R; //volume of circ. weld, m^3
```

```
            double CCircWeld = VCWeld * dens * Cw; //cost of circular
                  weld, $
            double CWeld = CLongWeld + CCircWeld; //total cost of weld

            double CTotal = CMat + CWeld; //total cost excluding
                  equipment cost, $
            return CTotal;
      } //end of Cost


      public double Rmin(double xV)
      {
            //find the minimum feasible radius for a vessel
            //with given Volume
            double V = xV;
            double Verr = 0.0; //used to compare with given V
            double R1 = 0.0;
            double R2 = Ru;
            double Rmin = 0.0; //minimum radius
            double error = 1.0; //error used for testing convergence
            //iterate through volume quation with bisection method
            for(int i=0; i<30; i++)
            {
                  Rmin = (R1+R2)/2;
                  Verr = (4/3)*pi*(Rmin*Rmin*Rmin) + pi*Rmin*Rmin*Lu;
                  error = java.lang.Math.abs(Verr - V);
                  if (error < 0.01) { break; }
                  if (Verr < V)
                  {
                        R1 = Rmin;
                  }
                  else
                  {
                        R2 = Rmin;
                  }
            }
            return Rmin;
      }//end of Rmin

      //end of FUNCTIONS

} //end of clsPressureVessel
```

clsResultOutput

```
/*
 * clsResultOutput.java
 */

 /* -- NOTES --
  * This class creates a BufferedWriter
  * it adds text with the addText() function
  * which requires an input of the nodeArray
  * it outputs a text file to be analyzed with Excel
 */
```

```
import java.io.*;

public class clsResultOutput
{
     BufferedWriter out;
     //---- Creates new clsResultOutput ----//
     public clsResultOutput(double res)
     {
           try
           {
                 out = new BufferedWriter(new FileWriter("Result_" +
                                          res + ".txt"));
                 out.write("Resolution = " + res);
                 out.write("\n");
           }
           catch (IOException e)
           {
                 System.err.println("FileStream error " + e);
           }
     }//end of clsResultOutput constructor


     //----FUNCTIONS----//

     //FUNCTION: addText//
     //function adds text to the opened file
     public void addText(double dV3, double dP2, double dV1, double
          dP1, double newprofit, double CEquip, double totalDemand,
          double newprofit1)
     {
           try
           {

     out.write("+++++++++++++++++++++++++++++++++++++++++++++");
                 out.write("\n");
                 out.write("dV3 = " + dV3 + "  dP2 = " + dP2 + "  dP1
                     = " + dP1 + "  dV1 = " + dV1);
                 out.write("\n");
                 out.write("initial profit = " + newprofit + "  CEquip
                      = " + CEquip + "  total demand = " +
                      totalDemand);
                 out.write("\n");
                 out.write("FINAL PROFIT = " + newprofit1);
                 out.write("\n");
           }
           catch (IOException e)
           {
                 System.err.println("FileStream error " + e);
           }
     }//end addText()

     //FUNCTION: close//
     //function closes file
     public void close()
     {
           try
```

```
            {
                  out.close();
            }
            catch (IOException e)
            {
                  System.err.println("FileStream error " + e);
            }
      }//end close()

}//end of clsResultOutput
```

APPENDIX B

MAPPING PRODUCT SPECIFICATIONS ONTO THE SPACE OF CUSTOMIZATION FOR EACH DEMAND SCENARIO OF THE PRESSURE VESSEL EXAMPLE

The results of applying the augmented PPCTM to each non-uniform demand scenario are presented in Table 4.6. The specific mapping of selected ranges for the modes of managing product customization to the product parameters is presented in Table 4.5. In this appendix, similar "roadmaps" are provided for each different demand scenario. These tables represent the application of the best set of ranges of managing variety for the specific demand scenario. From each "roadmap" the designer is able to quickly provide design specifications for pressure vessel for any customer requirement in the market space.

B.1 DISCRETE PYRAMID SMALL

This demand scenario models a "pyramid" of demand centered at the middle of both the volume and the pressure dimensions.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.3127918	3.6394052	
	11		8.04858	4.0037178	
	12		8.7851935	4.3680982	
	13		9.5226337	4.7325462	
	14		10.260902	5.0970621	
	15		11	5.4616457	
	16		11.739929	5.8262972	
	17		12.48069	6.1910165	
	18		13.222285	6.5558036	
	19		13.964715	6.9206585	
10	20	0.7832	14.707981	7.2855814	4.1449854
	21		15.452086	7.6505721	
	22		16.19703	8.0156308	
	23		16.942814	8.3807574	
	24		17.689441	8.745952	
	25		18.436911	9.1112145	
	26		19.185227	9.476545	
	27		19.934389	9.8419436	
	28		20.684399	10.20741	
	29		21.435259	10.572945	
	30		22.186969	10.938547	
V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.3127918	3.6394052	
	11		8.04858	4.0037178	
	12		8.7851935	4.3680982	
	13		9.5226337	4.7325462	
	14		10.260902	5.0970621	
	15		11	5.4616457	
	16		11.739929	5.8262972	
	17		12.48069	6.1910165	
	18		13.222285	6.5558036	
	19		13.964715	6.9206585	
15	20	0.7832	14.707981	7.2855814	6.7396115
	21		15.452086	7.6505721	
	22		16.19703	8.0156308	
	23		16.942814	8.3807574	
	24		17.689441	8.745952	
	25		18.436911	9.1112145	
	26		19.185227	9.476545	
	27		19.934389	9.8419436	
	28		20.684399	10.20741	
	29		21.435259	10.572945	
	30		22.186969	10.938547	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	
V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		9.3146592	4.6356877	
	11		10.251868	5.0997305	
	12		11.190129	5.5638595	
	13		12.129443	6.0280747	
	14		13.069811	6.4923763	
	15		14.011236	6.9567643	
	16		14.953719	7.4212386	
	17		15.897263	7.8857993	
	18		16.841868	8.3504464	
	19		17.787538	8.81518	
25	20	0.9976	18.734272	9.28	6.665949
	21		19.682074	9.7449065	
	22		20.630946	10.2099	
	23		21.580888	10.674979	
	24		22.531903	11.140145	
	25		23.483992	11.605398	
	26		24.437158	12.070737	
	27		25.391403	12.536163	
	28		26.346727	13.001676	
	29		27.303133	13.467275	
	30		28.260623	13.932961	

B.2 LINEAR LARGE

This scenario models demand as a linear function of volume and pressure. The maximum level of demand is 600 products.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		6.3053221	3.1380112	
	11		6.9397422	3.4521331	
	12		7.5748738	3.7663134	
	13		8.2107183	4.0805522	
	14		8.8472768	4.3948494	
	15		9.4845506	4.709205	
	16		10.122541	5.0236191	
	17		10.761249	5.3380917	
	18		11.400675	5.6526228	
	19		12.040822	5.9672124	
10	20	0.6753	12.68169	6.2818605	6.0796194
	21		13.323281	6.5965671	
	22		13.965595	6.9113323	
	23		14.608634	7.2261561	
	24		15.2524	7.5410385	
	25		15.896893	7.8559795	
	26		16.542114	8.1709792	
	27		17.188066	8.4860374	
	28		17.834748	8.8011543	
	29		18.482163	9.11633	
	30		19.130312	9.4315642	
	10		6.3053221	3.1380112	
	11		6.9397422	3.4521331	
	12		7.5748738	3.7663134	
	13		8.2107183	4.0805522	
	14		8.8472768	4.3948494	
	15		9.4845506	4.709205	
	16		10.122541	5.0236191	
	17		10.761249	5.3380917	
	18		11.400675	5.6526228	
	19		12.040822	5.9672124	
11	20	0.6753	12.68169	6.2818605	6.7776213
	21		13.323281	6.5965671	
	22		13.965595	6.9113323	
	23		14.608634	7.2261561	
	24		15.2524	7.5410385	
	25		15.896893	7.8559795	
	26		16.542114	8.1709792	
	27		17.188066	8.4860374	
	28		17.834748	8.8011543	
	29		18.482163	9.11633	
	30		19.130312	9.4315642	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
12	20	0.7041	13.222535	6.5497674	6.7660233
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	
	10		6.8295051	3.3988848	
	11		7.5166667	3.7391207	
	12		8.204599	4.07942	
	13		8.8933034	4.4197825	
	14		9.5827812	4.7602083	
	15		10.273034	5.1006974	
	16		10.964062	5.4412498	
	17		11.655868	5.7818655	
	18		12.348452	6.1225446	
	19		13.041817	6.4632871	
13	20	0.73144	13.735962	6.804093	6.7593147
	21		14.430891	7.1449623	
	22		15.126603	7.4858951	
	23		15.8231	7.8268912	
	24		16.520384	8.1679509	
	25		17.218456	8.509074	
	26		17.917317	8.8502606	
	27		18.616968	9.1915108	
	28		19.317412	9.5328244	
	29		20.018649	9.8742017	
	30		20.72068	10.215642	
	10		7.0756303	3.5213755	
	11		7.7875561	3.873873	
	12		8.5002804	4.2264361	
	13		9.2138047	4.5790648	
	14		9.9281303	4.931759	
	15		10.643258	5.2845188	
	16		11.359191	5.6373442	
	17		12.075928	5.9902353	
	18		12.793472	6.343192	
	19		13.511824	6.6962143	
14	20	0.7578	14.230986	7.0493023	6.7497294
	21		14.950958	7.402456	
	22		15.671743	7.7556755	
	23		16.393341	8.1089606	
	24		17.115754	8.4623116	
	25		17.838983	8.8157282	
	26		18.56303	9.1692107	
	27		19.287896	9.522759	
	28		20.013582	9.8763731	
	29		20.740091	10.230053	
	30		21.467422	10.583799	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.3127918	3.6394052	
	11		8.04858	4.0037178	
	12		8.7851935	4.3680982	
	13		9.5226337	4.7325462	
	14		10.260902	5.0970621	
	15		11	5.4616457	
	16		11.739929	5.8262972	
	17		12.48069	6.1910165	
	18		13.222285	6.5558036	
	19		13.964715	6.9206585	
15	20	0.7832	14.707981	7.2855814	6.7396115
	21		15.452086	7.6505721	
	22		16.19703	8.0156308	
	23		16.942814	8.3807574	
	24		17.689441	8.745952	
	25		18.436911	9.1112145	
	26		19.185227	9.476545	
	27		19.934389	9.8419436	
	28		20.684399	10.20741	
	29		21.435259	10.572945	
	30		22.186969	10.938547	
	10		7.5406162	3.7527881	
	11		8.2993274	4.1284506	
	12		9.0588895	4.5041829	
	13		9.8193042	4.8799851	
	14		10.580573	5.2558572	
	15		11.342697	5.6317992	
	16		12.105677	6.007811	
	17		12.869516	6.3838929	
	18		13.634215	6.7600446	
	19		14.399775	7.1362664	
16	20	0.8076	15.166197	7.5125581	6.7318775
	21		15.933484	7.8889199	
	22		16.701636	8.2653517	
	23		17.470655	8.6418535	
	24		18.240542	9.0184255	
	25		19.011299	9.3950675	
	26		19.782928	9.7717796	
	27		20.55543	10.148562	
	28		21.328806	10.525414	
	29		22.103058	10.902337	
	30		22.878187	11.27933	
	10		7.7591036	3.8615242	
	11		8.5397982	4.2480714	
	12		9.3213685	4.6346905	
	13		10.103816	5.0213814	
	14		10.887142	5.4081443	
	15		11.671348	5.7949791	
	16		12.456436	6.1818858	
	17		13.242407	6.5688645	
	18		14.029263	6.9559152	
	19		14.817005	7.3430379	
17	20	0.831	15.605634	7.7302326	6.7280464
	21		16.395152	8.1174993	
	22		17.185561	8.5048381	
	23		17.976862	8.892249	
	24		18.769057	9.279732	
	25		19.562147	9.6672871	
	26		20.356133	10.054914	
	27		21.151018	10.442614	
	28		21.946802	10.830385	
	29		22.743488	11.218229	
	30		23.541076	11.606145	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.9738562	3.9684015	
	11		8.7761584	4.3656474	
	12		9.5793606	4.7629671	
	13		10.383464	5.1603607	
	14		11.188471	5.5578282	
	15		11.994382	5.9553696	
	16		12.801199	6.3529849	
	17		13.608924	6.7506742	
	18		14.417558	7.1484375	
	19		15.227102	7.5462748	
18	20	0.854	16.037559	7.944186	6.7174313
	21		16.848929	8.3421714	
	22		17.661215	8.7402307	
	23		18.474417	9.1383642	
	24		19.288538	9.5365717	
	25		20.103578	9.9348534	
	26		20.91954	10.333209	
	27		21.736425	10.731639	
	28		22.554235	11.130143	
	29		23.372971	11.528722	
	30		24.192635	11.927374	
	10		8.1839402	4.0729554	
	11		9.0073804	4.4806673	
	12		9.8317443	4.8884551	
	13		10.657033	5.2963187	
	14		11.483249	5.7042581	
	15		12.310393	6.1122734	
	16		13.138467	6.5203645	
	17		13.967473	6.9285316	
	18		14.797411	7.3367746	
	19		15.628285	7.7450935	
19	20	0.8765	16.460094	8.1534884	6.7036016
	21		17.292841	8.5619593	
	22		18.126528	8.9705061	
	23		18.961155	9.3791291	
	24		19.796725	9.787828	
	25		20.633239	10.196603	
	26		21.470699	10.605454	
	27		22.309106	11.014381	
	28		23.148463	11.423385	
	29		23.988769	11.832464	
	30		24.830028	12.24162	
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.5798319	4.2699814	
	11		9.4431054	4.6974161	
	12		10.307347	5.1249303	
	13		11.172559	5.5525239	
	14		12.038742	5.9801971	
	15		12.905899	6.4079498	
	16		13.77403	6.835782	
	17		14.643138	7.2636939	
	18		15.513225	7.6916853	
	19		16.384291	8.1197563	
21	20	0.9189	17.256338	8.547907	6.6912946
	21		18.129369	8.9761373	
	22		19.003384	9.4044473	
	23		19.878386	9.8328371	
	24		20.754376	10.261307	
	25		21.631356	10.689856	
	26		22.509327	11.118485	
	27		23.388292	11.547194	
	28		24.268251	11.975982	
	29		25.149207	12.404851	
	30		26.031161	12.833799	
	10		8.7693744	4.3643123	
	11		9.651719	4.8011897	
	12		10.535053	5.2381484	
	13		11.419379	5.6751882	
	14		12.304698	6.1123094	
	15		13.191011	6.5495119	
	16		14.078321	6.9867956	
	17		14.966629	7.4241607	
	18		15.855937	7.8616071	
	19		16.746246	8.299135	
22	20	0.9392	17.637559	8.7367442	6.6865659
	21		18.529876	9.1744348	
	22		19.4232	9.6122069	
	23		20.317532	10.05006	
	24		21.212874	10.487996	
	25		22.109228	10.926012	
	26		23.006595	11.36411	
	27		23.904977	11.80229	
	28		24.804377	12.240551	
	29		25.704794	12.678894	
	30		26.606232	13.117318	
	10		8.9533147	4.455855	
	11		9.8541667	4.9018961	
	12		10.756029	5.3480201	
	13		11.658904	5.794227	
	14		12.562792	6.2405169	
	15		13.467697	6.6868898	
	16		14.373618	7.1333457	
	17		15.280559	7.5798847	
	18		16.18852	8.0265067	
	19		17.097504	8.4732118	
23	20	0.9589	18.007512	8.92	6.6836345
	21		18.918546	9.3668713	
	22		19.830607	9.8138258	
	23		20.743698	10.260863	
	24		21.65782	10.707984	
	25		22.572976	11.155188	
	26		23.489165	11.602476	
	27		24.406391	12.049846	
	28		25.324656	12.4973	
	29		26.24396	12.944838	
	30		27.164306	13.392458	

V	P	R (m)	Ts (mm)	Th (mm)	Lo
	10		9.136321	4.546933	
	11		10.05559	5.002091	
	12		10.97588	5.457334	
	13		11.89721	5.912662	
	14		12.81958	6.368074	
	15		13.74298	6.82357	
	16		14.66742	7.279152	
	17		15.59289	7.734818	
	18		16.51941	8.190569	
	19		17.44698	8.646405	
24	20	0.9785	18.37559	9.102326	6.674172
	21		19.30524	9.558331	
	22		20.23595	10.01442	
	23		21.1677	10.4706	
	24		22.10051	10.92686	
	25		23.03437	11.3832	
	26		23.96929	11.83963	
	27		24.90526	12.29615	
	28		25.84229	12.75275	
	29		26.78039	13.20943	
	30		27.71955	13.6662	
	10		9.314286	4.635502	
	11		10.25146	5.099526	
	12		11.18968	5.563636	
	13		12.12896	6.027833	
	14		13.06929	6.492116	
	15		14.01067	6.956485	
	16		14.95312	7.420941	
	17		15.89663	7.885483	
	18		16.84119	8.350112	
	19		17.78682	8.814827	
25	20	0.99756	18.73352	9.279628	6.666644
	21		19.68129	9.744516	
	22		20.63012	10.20949	
	23		21.58002	10.67455	
	24		22.531	11.1397	
	25		23.48305	11.60493	
	26		24.43618	12.07025	
	27		25.39038	12.53566	
	28		26.34567	13.00115	
	29		27.30204	13.46673	
	30		28.25949	13.9324	
	10		9.486461	4.72119	
	11		10.44096	5.193791	
	12		11.39652	5.666481	
	13		12.35316	6.139258	
	14		13.31087	6.612123	
	15		14.26966	7.085077	
	16		15.22953	7.558118	
	17		16.19048	8.031247	
	18		17.1525	8.504464	
	19		18.11562	8.97777	
26	20	1.016	19.07981	9.451163	6.66278
	21		20.0451	9.924644	
	22		21.01147	10.39821	
	23		21.97893	10.87187	
	24		22.94749	11.34562	
	25		23.91714	11.81945	
	26		24.88788	12.29337	
	27		25.85973	12.76738	
	28		26.83267	13.24148	
	29		27.80672	13.71567	
	30		28.78187	14.18994	

V	P	R (m)	Ts (mm)	Th (mm)	Lo
	10		9.654528	4.804833	
	11		10.62593	5.285807	
	12		11.59843	5.766871	
	13		12.57202	6.248025	
	14		13.5467	6.729267	
	15		14.52247	7.2106	
	16		15.49934	7.692022	
	17		16.47732	8.173533	
	18		17.45639	8.655134	
	19		18.43656	9.136824	
27	20	1.034	19.41784	9.618605	6.659793
	21		20.40023	10.10047	
	22		21.38372	10.58243	
	23		22.36832	11.06448	
	24		23.35404	11.54662	
	25		24.34087	12.02885	
	26		25.32881	12.51117	
	27		26.31787	12.99358	
	28		27.30806	13.47608	
	29		28.29936	13.95866	
	30		29.29178	14.44134	
	10		9.824463	4.889405	
	11		10.81297	5.378846	
	12		11.80258	5.868377	
	13		12.7933	6.357999	
	14		13.78514	6.847713	
	15		14.77809	7.337517	
	16		15.77216	7.827413	
	17		16.76734	8.3174	
	18		17.76365	8.807478	
	19		18.76107	9.297647	
28	20	1.0522	19.75962	9.787907	6.647357
	21		20.7593	10.27826	
	22		21.76011	10.7687	
	23		22.76204	11.25924	
	24		23.7651	11.74986	
	25		24.7693	12.24058	
	26		25.77464	12.73138	
	27		26.78111	13.22228	
	28		27.78872	13.71327	
	29		28.79747	14.20436	
	30		29.80737	14.69553	
	10		9.981326	4.967472	
	11		10.98561	5.464727	
	12		11.99103	5.962075	
	13		12.99757	6.459515	
	14		14.00524	6.957047	
	15		15.01404	7.454672	
	16		16.02398	7.95239	
	17		17.03506	8.4502	
	18		18.04727	8.948103	
	19		19.06062	9.446098	
29	20	1.069	20.07512	9.944186	6.65246
	21		21.09076	10.44237	
	22		22.10754	10.94064	
	23		23.12547	11.43901	
	24		24.14455	11.93747	
	25		25.16478	12.43602	
	26		26.18617	12.93466	
	27		27.20871	13.4334	
	28		28.23241	13.93223	
	29		29.25727	14.43115	
	30		30.28329	14.93017	

V	P	R (m)	Ts (mm)	Th (mm)	Lo
	10		10.14659	5.049721	
	11		11.16751	5.55521	
	12		12.18957	6.060792	
	13		13.21278	6.566468	
	14		14.23713	7.072239	
	15		15.26264	7.578103	
	16		16.2893	8.084062	
	17		17.31712	8.590114	
	18		18.34609	9.096261	
	19		19.37622	9.602502	
30	20	1.0867	20.40751	10.10884	6.637408
	21		21.43997	10.61527	
	22		22.47359	11.12179	
	23		23.50837	11.62841	
	24		24.54433	12.13512	
	25		25.58145	12.64193	
	26		26.61975	13.14883	
	27		27.65922	13.65582	
	28		28.69987	14.16291	
	29		29.74169	14.6701	
	30		30.7847	15.17737	

B.3 LINEAR SMALL

This scenario is identical to the previous, however the maximum demand is only 60 products.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.0840336	3.5255576	
	11		7.7968049	3.8784738	
	12		8.5103758	4.2314557	
	13		9.2247475	4.5845031	
	14		9.9399214	4.9376162	
	15		10.655899	5.290795	
	16		11.372681	5.6440394	
	17		12.09027	5.9973496	
	18		12.808666	6.3507254	
	19		13.527872	6.7041671	
10	20	0.7587	14.247887	7.0576744	4.5182068
	21		14.968715	7.4112476	
	22		15.690355	7.7648865	
	23		16.41281	8.1185912	
	24		17.136081	8.4723618	
	25		17.860169	8.8261982	
	26		18.585076	9.1801005	
	27		19.310803	9.5340687	
	28		20.037351	9.8881028	
	29		20.764723	10.242203	
	30		21.492918	10.596369	
	10		7.0840336	3.5255576	
	11		7.7968049	3.8784738	
	12		8.5103758	4.2314557	
	13		9.2247475	4.5845031	
	14		9.9399214	4.9376162	
	15		10.655899	5.290795	
	16		11.372681	5.6440394	
	17		12.09027	5.9973496	
	18		12.808666	6.3507254	
	19		13.527872	6.7041671	
14	20	0.7587	14.247887	7.0576744	6.7301295
	21		14.968715	7.4112476	
	22		15.690355	7.7648865	
	23		16.41281	8.1185912	
	24		17.136081	8.4723618	
	25		17.860169	8.8261982	
	26		18.585076	9.1801005	
	27		19.310803	9.5340687	
	28		20.037351	9.8881028	
	29		20.764723	10.242203	
	30		21.492918	10.596369	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.9738562	3.9684015	
	11		8.7761584	4.3656474	
	12		9.5793606	4.7629671	
	13		10.383464	5.1603607	
	14		11.188471	5.5578282	
	15		11.994382	5.9553696	
	16		12.801199	6.3529849	
	17		13.608924	6.7506742	
	18		14.417558	7.1484375	
	19		15.227102	7.5462748	
18	20	0.854	16.037559	7.944186	6.7174313
	21		16.848929	8.3421714	
	22		17.661215	8.7402307	
	23		18.474417	9.1383642	
	24		19.288538	9.5365717	
	25		20.103578	9.9348534	
	26		20.91954	10.333209	
	27		21.736425	10.731639	
	28		22.554235	11.130143	
	29		23.372971	11.528722	
	30		24.192635	11.927374	
	10		8.7693744	4.3643123	
	11		9.651719	4.8011897	
	12		10.535053	5.2381484	
	13		11.419379	5.6751882	
	14		12.304698	6.1123094	
	15		13.191011	6.5495119	
	16		14.078321	6.9867956	
	17		14.966629	7.4241607	
	18		15.855937	7.8616071	
	19		16.746246	8.299135	
22	20	0.9392	17.637559	8.7367442	6.6865659
	21		18.529876	9.1744348	
	22		19.4232	9.6122069	
	23		20.317532	10.05006	
	24		21.212874	10.487996	
	25		22.109228	10.926012	
	26		23.006595	11.36411	
	27		23.904977	11.80229	
	28		24.804377	12.240551	
	29		25.704794	12.678894	
	30		26.606232	13.117318	
	10		9.4864613	4.7211896	
	11		10.440957	5.1937912	
	12		11.396523	5.6664808	
	13		12.353161	6.1392582	
	14		13.310874	6.6121235	
	15		14.269663	7.0850767	
	16		15.22953	7.5581179	
	17		16.190476	8.0312471	
	18		17.152504	8.5044643	
	19		18.115616	8.9777695	
26	20	1.016	19.079812	9.4511628	6.6627796
	21		20.045096	9.9246442	
	22		21.011468	10.398214	
	23		21.978932	10.871871	
	24		22.947487	11.345617	
	25		23.917137	11.819451	
	26		24.887884	12.293373	
	27		25.859729	12.767383	
	28		26.832673	13.241482	
	29		27.80672	13.715669	
	30		28.78187	14.189944	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		10.146592	5.0497212	
	11		11.167507	5.5552096	
	12		12.189568	6.060792	
	13		13.212776	6.5664683	
	14		14.237133	7.0722388	
	15		15.26264	7.5781032	
	16		16.289301	8.0840617	
	17		17.317117	8.5901144	
	18		18.346089	9.0962612	
	19		19.37622	9.6025021	
30	20	1.0867	20.407512	10.108837	6.6374076
	21		21.439966	10.615267	
	22		22.473585	11.12179	
	23		23.508371	11.628408	
	24		24.544325	12.13512	
	25		25.58145	12.641926	
	26		26.619748	13.148827	
	27		27.659219	13.655822	
	28		28.699868	14.162912	
	29		29.741695	14.670096	
	30		30.784703	15.177374	

B.4 NORMAL CENTER SMALL

This demand scenario approximates the demand of the market as a two-dimensional Gaussian distribution, with the mean located at the center of the space of customization.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
10	20	0.7041	13.222535	6.5497674	5.4818861
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
12	20	0.7041	13.222535	6.5497674	6.7660233
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.0756303	3.5213755	
	11		7.7875561	3.873873	
	12		8.5002804	4.2264361	
	13		9.2138047	4.5790648	
	14		9.9281303	4.931759	
	15		10.643258	5.2845188	
	16		11.359191	5.6373442	
	17		12.075928	5.9902353	
	18		12.793472	6.343192	
	19		13.511824	6.6962143	
14	20	0.7578	14.230986	7.0493023	6.7497294
	21		14.950958	7.402456	
	22		15.671743	7.7556755	
	23		16.393341	8.1089606	
	24		17.115754	8.4623116	
	25		17.838983	8.8157282	
	26		18.56303	9.1692107	
	27		19.287896	9.522759	
	28		20.013582	9.8763731	
	29		20.740091	10.230053	
	30		21.467422	10.583799	
	10		7.5406162	3.7527881	
	11		8.2993274	4.1284506	
	12		9.0588895	4.5041829	
	13		9.8193042	4.8799851	
	14		10.580573	5.2558572	
	15		11.342697	5.6317992	
	16		12.105677	6.007811	
	17		12.869516	6.3838929	
	18		13.634215	6.7600446	
	19		14.399775	7.1362664	
16	20	0.8076	15.166197	7.5125581	6.7318775
	21		15.933484	7.8889199	
	22		16.701636	8.2653517	
	23		17.470655	8.6418535	
	24		18.240542	9.0184255	
	25		19.011299	9.3950675	
	26		19.782928	9.7717796	
	27		20.55543	10.148562	
	28		21.328806	10.525414	
	29		22.103058	10.902337	
	30		22.878187	11.27933	
	10		7.9738562	3.9684015	
	11		8.7761584	4.3656474	
	12		9.5793606	4.7629671	
	13		10.383464	5.1603607	
	14		11.188471	5.5578282	
	15		11.994382	5.9553696	
	16		12.801199	6.3529849	
	17		13.608924	6.7506742	
	18		14.417558	7.1484375	
	19		15.227102	7.5462748	
18	20	0.854	16.037559	7.944186	6.7174313
	21		16.848929	8.3421714	
	22		17.661215	8.7402307	
	23		18.474417	9.1383642	
	24		19.288538	9.5365717	
	25		20.103578	9.9348534	
	26		20.91954	10.333209	
	27		21.736425	10.731639	
	28		22.554235	11.130143	
	29		23.372971	11.528722	
	30		24.192635	11.927374	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	
	10		8.7693744	4.3643123	
	11		9.651719	4.8011897	
	12		10.535053	5.2381484	
	13		11.419379	5.6751882	
	14		12.304698	6.1123094	
	15		13.191011	6.5495119	
	16		14.078321	6.9867956	
	17		14.966629	7.4241607	
	18		15.855937	7.8616071	
	19		16.746246	8.299135	
22	20	0.9392	17.637559	8.7367442	6.6865659
	21		18.529876	9.1744348	
	22		19.4232	9.6122069	
	23		20.317532	10.05006	
	24		21.212874	10.487996	
	25		22.109228	10.926012	
	26		23.006595	11.36411	
	27		23.904977	11.80229	
	28		24.804377	12.240551	
	29		25.704794	12.678894	
	30		26.606232	13.117318	
	10		9.1363212	4.5469331	
	11		10.055587	5.0020913	
	12		10.975883	5.4573341	
	13		11.897213	5.9126615	
	14		12.819577	6.3680736	
	15		13.742978	6.8235704	
	16		14.667416	7.2791519	
	17		15.592895	7.7348182	
	18		16.519415	8.1905692	
	19		17.446978	8.646405	
24	20	0.9785	18.375587	9.1023256	6.6741725
	21		19.305242	9.558331	
	22		20.235947	10.014421	
	23		21.167701	10.470596	
	24		22.100508	10.926857	
	25		23.034369	11.383201	
	26		23.969286	11.839631	
	27		24.90526	12.296146	
	28		25.842294	12.752746	
	29		26.780389	13.209431	
	30		27.719547	13.666201	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		9.4864613	4.7211896	
	11		10.440957	5.1937912	
	12		11.396523	5.6664808	
	13		12.353161	6.1392582	
	14		13.310874	6.6121235	
	15		14.269663	7.0850767	
	16		15.22953	7.5581179	
	17		16.190476	8.0312471	
	18		17.152504	8.5044643	
	19		18.115616	8.9777695	
26	20	1.016	19.079812	9.4511628	6.6627796
	21		20.045096	9.9246442	
	22		21.011468	10.398214	
	23		21.978932	10.871871	
	24		22.947487	11.345617	
	25		23.917137	11.819451	
	26		24.887884	12.293373	
	27		25.859729	12.767383	
	28		26.832673	13.241482	
	29		27.80672	13.715669	
	30		28.78187	14.189944	
	10		9.8244631	4.8894052	
	11		10.812967	5.3788456	
	12		11.80258	5.868377	
	13		12.793303	6.3579994	
	14		13.785139	6.8477129	
	15		14.77809	7.3375174	
	16		15.772157	7.8274131	
	17		16.767342	8.3173998	
	18		17.763647	8.8074777	
	19		18.761074	9.2976467	
28	20	1.0522	19.759624	9.787907	6.6473574
	21		20.759301	10.278258	
	22		21.760105	10.768701	
	23		22.762039	11.259235	
	24		23.765104	11.74986	
	25		24.769303	12.240577	
	26		25.774637	12.731385	
	27		26.781109	13.222284	
	28		27.788719	13.713275	
	29		28.797471	14.204357	
	30		29.807365	14.695531	
	10		10.145658	5.0492565	
	11		11.16648	5.5546984	
	12		12.188446	6.0602342	
	13		13.21156	6.5658641	
	14		14.235823	7.071588	
	15		15.261236	7.5774059	
	16		16.287802	8.0833178	
	17		17.315523	8.5893239	
	18		18.344401	9.0954241	
	19		19.374437	9.6016185	
30	20	1.0866	20.405634	10.107907	6.6390294
	21		21.437993	10.61429	
	22		22.471517	11.120767	
	23		23.506208	11.627338	
	24		24.542067	12.134003	
	25		25.579096	12.640763	
	26		26.617298	13.147617	
	27		27.656674	13.654566	
	28		28.697227	14.161609	
	29		29.738958	14.668746	
	30		30.78187	15.175978	

B.5 NORMAL LOW

This scenario is similar to the previous scenario except that the mean is placed near the lower-end vessels of the market space.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
10	20	0.7041	13.222535	6.5497674	5.4818861
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
12	20	0.7041	13.222535	6.5497674	6.7660233
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.0756303	3.5213755	
	11		7.7875561	3.873873	
	12		8.5002804	4.2264361	
	13		9.2138047	4.5790648	
	14		9.9281303	4.931759	
	15		10.643258	5.2845188	
	16		11.359191	5.6373442	
	17		12.075928	5.9902353	
	18		12.793472	6.343192	
	19		13.511824	6.6962143	
14	20	0.7578	14.230986	7.0493023	6.7497294
	21		14.950958	7.402456	
	22		15.671743	7.7556755	
	23		16.393341	8.1089606	
	24		17.115754	8.4623116	
	25		17.838983	8.8157282	
	26		18.56303	9.1692107	
	27		19.287896	9.522759	
	28		20.013582	9.8763731	
	29		20.740091	10.230053	
	30		21.467422	10.583799	
	10		7.5406162	3.7527881	
	11		8.2993274	4.1284506	
	12		9.0588895	4.5041829	
	13		9.8193042	4.8799851	
	14		10.580573	5.2558572	
	15		11.342697	5.6317992	
	16		12.105677	6.007811	
	17		12.869516	6.3838929	
	18		13.634215	6.7600446	
	19		14.399775	7.1362664	
16	20	0.8076	15.166197	7.5125581	6.7318775
	21		15.933484	7.8889199	
	22		16.701636	8.2653517	
	23		17.470655	8.6418535	
	24		18.240542	9.0184255	
	25		19.011299	9.3950675	
	26		19.782928	9.7717796	
	27		20.55543	10.148562	
	28		21.328806	10.525414	
	29		22.103058	10.902337	
	30		22.878187	11.27933	
	10		7.9738562	3.9684015	
	11		8.7761584	4.3656474	
	12		9.5793606	4.7629671	
	13		10.383464	5.1603607	
	14		11.188471	5.5578282	
	15		11.994382	5.9553696	
	16		12.801199	6.3529849	
	17		13.608924	6.7506742	
	18		14.417558	7.1484375	
	19		15.227102	7.5462748	
18	20	0.854	16.037559	7.944186	6.7174313
	21		16.848929	8.3421714	
	22		17.661215	8.7402307	
	23		18.474417	9.1383642	
	24		19.288538	9.5365717	
	25		20.103578	9.9348534	
	26		20.91954	10.333209	
	27		21.736425	10.731639	
	28		22.554235	11.130143	
	29		23.372971	11.528722	
	30		24.192635	11.927374	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	
	10		8.7693744	4.3643123	
	11		9.651719	4.8011897	
	12		10.535053	5.2381484	
	13		11.419379	5.6751882	
	14		12.304698	6.1123094	
	15		13.191011	6.5495119	
	16		14.078321	6.9867956	
	17		14.966629	7.4241607	
	18		15.855937	7.8616071	
	19		16.746246	8.299135	
22	20	0.9392	17.637559	8.7367442	6.6865659
	21		18.529876	9.1744348	
	22		19.4232	9.6122069	
	23		20.317532	10.05006	
	24		21.212874	10.487996	
	25		22.109228	10.926012	
	26		23.006595	11.36411	
	27		23.904977	11.80229	
	28		24.804377	12.240551	
	29		25.704794	12.678894	
	30		26.606232	13.117318	

B.6 NORMAL HIGH

This scenario is extremely similar to the previous scenario, except the mean of the distribution is located near the higher-end pressure vessels.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
10	10	0.7041	6.5742297	3.2718401	5.4818861
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
	20		13.222535	6.5497674	
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	
12	10	0.7041	6.5742297	3.2718401	6.7660233
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
	20		13.222535	6.5497674	
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.0756303	3.5213755	
	11		7.7875561	3.873873	
	12		8.5002804	4.2264361	
	13		9.2138047	4.5790648	
	14		9.9281303	4.931759	
	15		10.643258	5.2845188	
	16		11.359191	5.6373442	
	17		12.075928	5.9902353	
	18		12.793472	6.343192	
	19		13.511824	6.6962143	
14	20	0.7578	14.230986	7.0493023	6.7497294
	21		14.950958	7.402456	
	22		15.671743	7.7556755	
	23		16.393341	8.1089606	
	24		17.115754	8.4623116	
	25		17.838983	8.8157282	
	26		18.56303	9.1692107	
	27		19.287896	9.522759	
	28		20.013582	9.8763731	
	29		20.740091	10.230053	
	30		21.467422	10.583799	
	10		7.5406162	3.7527881	
	11		8.2993274	4.1284506	
	12		9.0588895	4.5041829	
	13		9.8193042	4.8799851	
	14		10.580573	5.2558572	
	15		11.342697	5.6317992	
	16		12.105677	6.007811	
	17		12.869516	6.3838929	
	18		13.634215	6.7600446	
	19		14.399775	7.1362664	
16	20	0.8076	15.166197	7.5125581	6.7318775
	21		15.933484	7.8889199	
	22		16.701636	8.2653517	
	23		17.470655	8.6418535	
	24		18.240542	9.0184255	
	25		19.011299	9.3950675	
	26		19.782928	9.7717796	
	27		20.55543	10.148562	
	28		21.328806	10.525414	
	29		22.103058	10.902337	
	30		22.878187	11.27933	
	10		7.9738562	3.9684015	
	11		8.7761584	4.3656474	
	12		9.5793606	4.7629671	
	13		10.383464	5.1603607	
	14		11.188471	5.5578282	
	15		11.994382	5.9553696	
	16		12.801199	6.3529849	
	17		13.608924	6.7506742	
	18		14.417558	7.1484375	
	19		15.227102	7.5462748	
18	20	0.854	16.037559	7.944186	6.7174313
	21		16.848929	8.3421714	
	22		17.661215	8.7402307	
	23		18.474417	9.1383642	
	24		19.288538	9.5365717	
	25		20.103578	9.9348534	
	26		20.91954	10.333209	
	27		21.736425	10.731639	
	28		22.554235	11.130143	
	29		23.372971	11.528722	
	30		24.192635	11.927374	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	
	10		8.7693744	4.3643123	
	11		9.651719	4.8011897	
	12		10.535053	5.2381484	
	13		11.419379	5.6751882	
	14		12.304698	6.1123094	
	15		13.191011	6.5495119	
	16		14.078321	6.9867956	
	17		14.966629	7.4241607	
	18		15.855937	7.8616071	
	19		16.746246	8.299135	
22	20	0.9392	17.637559	8.7367442	6.6865659
	21		18.529876	9.1744348	
	22		19.4232	9.6122069	
	23		20.317532	10.05006	
	24		21.212874	10.487996	
	25		22.109228	10.926012	
	26		23.006595	11.36411	
	27		23.904977	11.80229	
	28		24.804377	12.240551	
	29		25.704794	12.678894	
	30		26.606232	13.117318	
	10		9.1363212	4.5469331	
	11		10.055587	5.0020913	
	12		10.975883	5.4573341	
	13		11.897213	5.9126615	
	14		12.819577	6.3680736	
	15		13.742978	6.8235704	
	16		14.667416	7.2791519	
	17		15.592895	7.7348182	
	18		16.519415	8.1905692	
	19		17.446978	8.646405	
24	20	0.9785	18.375587	9.1023256	6.6741725
	21		19.305242	9.558331	
	22		20.235947	10.014421	
	23		21.167701	10.470596	
	24		22.100508	10.926857	
	25		23.034369	11.383201	
	26		23.969286	11.839631	
	27		24.90526	12.296146	
	28		25.842294	12.752746	
	29		26.780389	13.209431	
	30		27.719547	13.666201	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		9.4864613	4.7211896	
	11		10.440957	5.1937912	
	12		11.396523	5.6664808	
	13		12.353161	6.1392582	
	14		13.310874	6.6121235	
	15		14.269663	7.0850767	
	16		15.22953	7.5581179	
	17		16.190476	8.0312471	
	18		17.152504	8.5044643	
	19		18.115616	8.9777695	
26	20	1.016	19.079812	9.4511628	6.6627796
	21		20.045096	9.9246442	
	22		21.011468	10.398214	
	23		21.978932	10.871871	
	24		22.947487	11.345617	
	25		23.917137	11.819451	
	26		24.887884	12.293373	
	27		25.859729	12.767383	
	28		26.832673	13.241482	
	29		27.80672	13.715669	
	30		28.78187	14.189944	
	10		9.8225957	4.8884758	
	11		10.810912	5.3778232	
	12		11.800337	5.8672616	
	13		12.790872	6.3567909	
	14		13.782519	6.8464113	
	15		14.775281	7.3361227	
	16		15.769159	7.8259252	
	17		16.764154	8.3158188	
	18		17.76027	8.8058036	
	19		18.757508	9.2958795	
28	20	1.052	19.755869	9.7860465	6.6506853
	21		20.755355	10.276305	
	22		21.755969	10.766654	
	23		22.757713	11.257095	
	24		23.760587	11.747627	
	25		24.764595	12.23825	
	26		25.769738	12.728965	
	27		26.776018	13.219771	
	28		27.783437	13.710668	
	29		28.791997	14.201657	
	30		29.8017	14.692737	
	10		10.146592	5.0497212	
	11		11.167507	5.5552096	
	12		12.189568	6.060792	
	13		13.212776	6.5664683	
	14		14.237133	7.0722388	
	15		15.26264	7.5781032	
	16		16.289301	8.0840617	
	17		17.317117	8.5901144	
	18		18.346089	9.0962612	
	19		19.37622	9.6025021	
30	20	1.0867	20.407512	10.108837	6.6374076
	21		21.439966	10.615267	
	22		22.473585	11.12179	
	23		23.508371	11.628408	
	24		24.544325	12.13512	
	25		25.58145	12.641926	
	26		26.619748	13.148827	
	27		27.659219	13.655822	
	28		28.699868	14.162912	
	29		29.741695	14.670096	
	30		30.784703	15.177374	

B.7 NORMAL CENTER LARGE

Similar to "Normal Center Small," this demand scenario places the majority of the demand in the center of the customization space. This demand model, however, approximates a much larger amount of products.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
10	20	0.7041	13.222535	6.5497674	5.4818861
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
12	20	0.7041	13.222535	6.5497674	6.7660233
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.0756303	3.5213755	
	11		7.7875561	3.873873	
	12		8.5002804	4.2264361	
	13		9.2138047	4.5790648	
	14		9.9281303	4.931759	
	15		10.643258	5.2845188	
	16		11.359191	5.6373442	
	17		12.075928	5.9902353	
	18		12.793472	6.343192	
	19		13.511824	6.6962143	
14	20	0.7578	14.230986	7.0493023	6.7497294
	21		14.950958	7.402456	
	22		15.671743	7.7556755	
	23		16.393341	8.1089606	
	24		17.115754	8.4623116	
	25		17.838983	8.8157282	
	26		18.56303	9.1692107	
	27		19.287896	9.522759	
	28		20.013582	9.8763731	
	29		20.740091	10.230053	
	30		21.467422	10.583799	
	10		7.5406162	3.7527881	
	11		8.2993274	4.1284506	
	12		9.0588895	4.5041829	
	13		9.8193042	4.8799851	
	14		10.580573	5.2558572	
	15		11.342697	5.6317992	
	16		12.105677	6.007811	
	17		12.869516	6.3838929	
	18		13.634215	6.7600446	
	19		14.399775	7.1362664	
16	20	0.8076	15.166197	7.5125581	6.7318775
	21		15.933484	7.8889199	
	22		16.701636	8.2653517	
	23		17.470655	8.6418535	
	24		18.240542	9.0184255	
	25		19.011299	9.3950675	
	26		19.782928	9.7717796	
	27		20.55543	10.148562	
	28		21.328806	10.525414	
	29		22.103058	10.902337	
	30		22.878187	11.27933	
	10		7.9738562	3.9684015	
	11		8.7761584	4.3656474	
	12		9.5793606	4.7629671	
	13		10.383464	5.1603607	
	14		11.188471	5.5578282	
	15		11.994382	5.9553696	
	16		12.801199	6.3529849	
	17		13.608924	6.7506742	
	18		14.417558	7.1484375	
	19		15.227102	7.5462748	
18	20	0.854	16.037559	7.944186	6.7174313
	21		16.848929	8.3421714	
	22		17.661215	8.7402307	
	23		18.474417	9.1383642	
	24		19.288538	9.5365717	
	25		20.103578	9.9348534	
	26		20.91954	10.333209	
	27		21.736425	10.731639	
	28		22.554235	11.130143	
	29		23.372971	11.528722	
	30		24.192635	11.927374	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	
	10		8.7693744	4.3643123	
	11		9.651719	4.8011897	
	12		10.535053	5.2381484	
	13		11.419379	5.6751882	
	14		12.304698	6.1123094	
	15		13.191011	6.5495119	
	16		14.078321	6.9867956	
	17		14.966629	7.4241607	
	18		15.855937	7.8616071	
	19		16.746246	8.299135	
22	20	0.9392	17.637559	8.7367442	6.6865659
	21		18.529876	9.1744348	
	22		19.4232	9.6122069	
	23		20.317532	10.05006	
	24		21.212874	10.487996	
	25		22.109228	10.926012	
	26		23.006595	11.36411	
	27		23.904977	11.80229	
	28		24.804377	12.240551	
	29		25.704794	12.678894	
	30		26.606232	13.117318	
	10		9.1363212	4.5469331	
	11		10.055587	5.0020913	
	12		10.975883	5.4573341	
	13		11.897213	5.9126615	
	14		12.819577	6.3680736	
	15		13.742978	6.8235704	
	16		14.667416	7.2791519	
	17		15.592895	7.7348182	
	18		16.519415	8.1905692	
	19		17.446978	8.646405	
24	20	0.9785	18.375587	9.1023256	6.6741725
	21		19.305242	9.558331	
	22		20.235947	10.014421	
	23		21.167701	10.470596	
	24		22.100508	10.926857	
	25		23.034369	11.383201	
	26		23.969286	11.839631	
	27		24.90526	12.296146	
	28		25.842294	12.752746	
	29		26.780389	13.209431	
	30		27.719547	13.666201	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		9.4864613	4.7211896	
	11		10.440957	5.1937912	
	12		11.396523	5.6664808	
	13		12.353161	6.1392582	
	14		13.310874	6.6121235	
	15		14.269663	7.0850767	
	16		15.22953	7.5581179	
	17		16.190476	8.0312471	
	18		17.152504	8.5044643	
	19		18.115616	8.9777695	
26	20	1.016	19.079812	9.4511628	6.6627796
	21		20.045096	9.9246442	
	22		21.011468	10.398214	
	23		21.978932	10.871871	
	24		22.947487	11.345617	
	25		23.917137	11.819451	
	26		24.887884	12.293373	
	27		25.859729	12.767383	
	28		26.832673	13.241482	
	29		27.80672	13.715669	
	30		28.78187	14.189944	

B.8 RANDOM LARGE

This demand scenario models a large random distribution of demand.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
10	10	0.6753	6.3053221	3.1380112	6.0796194
	11		6.9397422	3.4521331	
	12		7.5748738	3.7663134	
	13		8.2107183	4.0805522	
	14		8.8472768	4.3948494	
	15		9.4845506	4.709205	
	16		10.122541	5.0236191	
	17		10.761249	5.3380917	
	18		11.400675	5.6526228	
	19		12.040822	5.9672124	
	20		12.68169	6.2818605	
	21		13.323281	6.5965671	
	22		13.965595	6.9113323	
	23		14.608634	7.2261561	
	24		15.2524	7.5410385	
	25		15.896893	7.8559795	
	26		16.542114	8.1709792	
	27		17.188066	8.4860374	
	28		17.834748	8.8011543	
	29		18.482163	9.11633	
	30		19.130312	9.4315642	
11	10	0.6753	6.3053221	3.1380112	6.7776213
	11		6.9397422	3.4521331	
	12		7.5748738	3.7663134	
	13		8.2107183	4.0805522	
	14		8.8472768	4.3948494	
	15		9.4845506	4.709205	
	16		10.122541	5.0236191	
	17		10.761249	5.3380917	
	18		11.400675	5.6526228	
	19		12.040822	5.9672124	
	20		12.68169	6.2818605	
	21		13.323281	6.5965671	
	22		13.965595	6.9113323	
	23		14.608634	7.2261561	
	24		15.2524	7.5410385	
	25		15.896893	7.8559795	
	26		16.542114	8.1709792	
	27		17.188066	8.4860374	
	28		17.834748	8.8011543	
	29		18.482163	9.11633	
	30		19.130312	9.4315642	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		6.5742297	3.2718401	
	11		7.2357063	3.5993587	
	12		7.8979248	3.9269381	
	13		8.5608866	4.2545784	
	14		9.2245929	4.5822797	
	15		9.8890449	4.9100418	
	16		10.554244	5.237865	
	17		11.220191	5.5657491	
	18		11.886888	5.8936942	
	19		12.554336	6.2217003	
12	20	0.7041	13.222535	6.5497674	6.7660233
	21		13.891488	6.8778956	
	22		14.561196	7.2060849	
	23		15.231659	7.5343352	
	24		15.90288	7.8626466	
	25		16.574859	8.1910191	
	26		17.247598	8.5194527	
	27		17.921097	8.8479475	
	28		18.595359	9.1765034	
	29		19.270385	9.5051206	
	30		19.946176	9.8337989	
	10		6.8295051	3.3988848	
	11		7.5166667	3.7391207	
	12		8.204599	4.07942	
	13		8.8933034	4.4197825	
	14		9.5827812	4.7602083	
	15		10.273034	5.1006974	
	16		10.964062	5.4412498	
	17		11.655868	5.7818655	
	18		12.348452	6.1225446	
	19		13.041817	6.4632871	
13	20	0.73144	13.735962	6.804093	6.7593147
	21		14.430891	7.1449623	
	22		15.126603	7.4858951	
	23		15.8231	7.8268912	
	24		16.520384	8.1679509	
	25		17.218456	8.509074	
	26		17.917317	8.8502606	
	27		18.616968	9.1915108	
	28		19.317412	9.5328244	
	29		20.018649	9.8742017	
	30		20.72068	10.215642	
	10		7.0756303	3.5213755	
	11		7.7875561	3.873873	
	12		8.5002804	4.2264361	
	13		9.2138047	4.5790648	
	14		9.9281303	4.931759	
	15		10.643258	5.2845188	
	16		11.359191	5.6373442	
	17		12.075928	5.9902353	
	18		12.793472	6.343192	
	19		13.511824	6.6962143	
14	20	0.7578	14.230986	7.0493023	6.7497294
	21		14.950958	7.402456	
	22		15.671743	7.7556755	
	23		16.393341	8.1089606	
	24		17.115754	8.4623116	
	25		17.838983	8.8157282	
	26		18.56303	9.1692107	
	27		19.287896	9.522759	
	28		20.013582	9.8763731	
	29		20.740091	10.230053	
	30		21.467422	10.583799	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.3127918	3.6394052	
	11		8.04858	4.0037178	
	12		8.7851935	4.3680982	
	13		9.5226337	4.7325462	
	14		10.260902	5.0970621	
	15		11	5.4616457	
	16		11.739929	5.8262972	
	17		12.48069	6.1910165	
	18		13.222285	6.5558036	
	19		13.964715	6.9206585	
15	20	0.7832	14.707981	7.2855814	6.7396115
	21		15.452086	7.6505721	
	22		16.19703	8.0156308	
	23		16.942814	8.3807574	
	24		17.689441	8.745952	
	25		18.436911	9.1112145	
	26		19.185227	9.476545	
	27		19.934389	9.8419436	
	28		20.684399	10.20741	
	29		21.435259	10.572945	
	30		22.186969	10.938547	
	10		7.5406162	3.7527881	
	11		8.2993274	4.1284506	
	12		9.0588895	4.5041829	
	13		9.8193042	4.8799851	
	14		10.580573	5.2558572	
	15		11.342697	5.6317992	
	16		12.105677	6.007811	
	17		12.869516	6.3838929	
	18		13.634215	6.7600446	
	19		14.399775	7.1362664	
16	20	0.8076	15.166197	7.5125581	6.7318775
	21		15.933484	7.8889199	
	22		16.701636	8.2653517	
	23		17.470655	8.6418535	
	24		18.240542	9.0184255	
	25		19.011299	9.3950675	
	26		19.782928	9.7717796	
	27		20.55543	10.148562	
	28		21.328806	10.525414	
	29		22.103058	10.902337	
	30		22.878187	11.27933	
	10		7.7591036	3.8615242	
	11		8.5397982	4.2480714	
	12		9.3213685	4.6346905	
	13		10.103816	5.0213814	
	14		10.887142	5.4081443	
	15		11.671348	5.7949791	
	16		12.456436	6.1818858	
	17		13.242407	6.5688645	
	18		14.029263	6.9559152	
	19		14.817005	7.3430379	
17	20	0.831	15.605634	7.7302326	6.7280464
	21		16.395152	8.1174993	
	22		17.185561	8.5048381	
	23		17.976862	8.892249	
	24		18.769057	9.279732	
	25		19.562147	9.6672871	
	26		20.356133	10.054914	
	27		21.151018	10.442614	
	28		21.946802	10.830385	
	29		22.743488	11.218229	
	30		23.541076	11.606145	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		7.9738562	3.9684015	
	11		8.7761584	4.3656474	
	12		9.5793606	4.7629671	
	13		10.383464	5.1603607	
	14		11.188471	5.5578282	
	15		11.994382	5.9553696	
	16		12.801199	6.3529849	
	17		13.608924	6.7506742	
	18		14.417558	7.1484375	
	19		15.227102	7.5462748	
18	20	0.854	16.037559	7.944186	6.7174313
	21		16.848929	8.3421714	
	22		17.661215	8.7402307	
	23		18.474417	9.1383642	
	24		19.288538	9.5365717	
	25		20.103578	9.9348534	
	26		20.91954	10.333209	
	27		21.736425	10.731639	
	28		22.554235	11.130143	
	29		23.372971	11.528722	
	30		24.192635	11.927374	
	10		8.1839402	4.0729554	
	11		9.0073804	4.4806673	
	12		9.8317443	4.8884551	
	13		10.657033	5.2963187	
	14		11.483249	5.7042581	
	15		12.310393	6.1122734	
	16		13.138467	6.5203645	
	17		13.967473	6.9285316	
	18		14.797411	7.3367746	
	19		15.628285	7.7450935	
19	20	0.8765	16.460094	8.1534884	6.7036016
	21		17.292841	8.5619593	
	22		18.126528	8.9705061	
	23		18.961155	9.3791291	
	24		19.796725	9.787828	
	25		20.633239	10.196603	
	26		21.470699	10.605454	
	27		22.309106	11.014381	
	28		23.148463	11.423385	
	29		23.988769	11.832464	
	30		24.830028	12.24162	
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.5798319	4.2699814	
	11		9.4431054	4.6974161	
	12		10.307347	5.1249303	
	13		11.172559	5.5525239	
	14		12.038742	5.9801971	
	15		12.905899	6.4079498	
	16		13.77403	6.835782	
	17		14.643138	7.2636939	
	18		15.513225	7.6916853	
	19		16.384291	8.1197563	
21	20	0.9189	17.256338	8.547907	6.6912946
	21		18.129369	8.9761373	
	22		19.003384	9.4044473	
	23		19.878386	9.8328371	
	24		20.754376	10.261307	
	25		21.631356	10.689856	
	26		22.509327	11.118485	
	27		23.388292	11.547194	
	28		24.268251	11.975982	
	29		25.149207	12.404851	
	30		26.031161	12.833799	
	10		8.7693744	4.3643123	
	11		9.651719	4.8011897	
	12		10.535053	5.2381484	
	13		11.419379	5.6751882	
	14		12.304698	6.1123094	
	15		13.191011	6.5495119	
	16		14.078321	6.9867956	
	17		14.966629	7.4241607	
	18		15.855937	7.8616071	
	19		16.746246	8.299135	
22	20	0.9392	17.637559	8.7367442	6.6865659
	21		18.529876	9.1744348	
	22		19.4232	9.6122069	
	23		20.317532	10.05006	
	24		21.212874	10.487996	
	25		22.109228	10.926012	
	26		23.006595	11.36411	
	27		23.904977	11.80229	
	28		24.804377	12.240551	
	29		25.704794	12.678894	
	30		26.606232	13.117318	
	10		8.9533147	4.455855	
	11		9.8541667	4.9018961	
	12		10.756029	5.3480201	
	13		11.658904	5.794227	
	14		12.562792	6.2405169	
	15		13.467697	6.6868898	
	16		14.373618	7.1333457	
	17		15.280559	7.5798847	
	18		16.18852	8.0265067	
	19		17.097504	8.4732118	
23	20	0.9589	18.007512	8.92	6.6836345
	21		18.918546	9.3668713	
	22		19.830607	9.8138258	
	23		20.743698	10.260863	
	24		21.65782	10.707984	
	25		22.572976	11.155188	
	26		23.489165	11.602476	
	27		24.406391	12.049846	
	28		25.324656	12.4973	
	29		26.24396	12.944838	
	30		27.164306	13.392458	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		9.1363212	4.5469331	
	11		10.055587	5.0020913	
	12		10.975883	5.4573341	
	13		11.897213	5.9126615	
	14		12.819577	6.3680736	
	15		13.742978	6.8235704	
	16		14.667416	7.2791519	
	17		15.592895	7.7348182	
	18		16.519415	8.1905692	
	19		17.446978	8.646405	
24	20	0.9785	18.375587	9.1023256	6.6741725
	21		19.305242	9.558331	
	22		20.235947	10.014421	
	23		21.167701	10.470596	
	24		22.100508	10.926857	
	25		23.034369	11.383201	
	26		23.969286	11.839631	
	27		24.90526	12.296146	
	28		25.842294	12.752746	
	29		26.780389	13.209431	
	30		27.719547	13.666201	
	10		9.3142857	4.6355019	
	11		10.251457	5.099526	
	12		11.18968	5.5636364	
	13		12.128956	6.027833	
	14		13.069287	6.492116	
	15		14.010674	6.9564854	
	16		14.95312	7.420941	
	17		15.896625	7.8854831	
	18		16.841193	8.3501116	
	19		17.786824	8.8148265	
25	20	0.99756	18.733521	9.2796279	6.6666436
	21		19.681285	9.7445158	
	22		20.630118	10.20949	
	23		21.580023	10.674551	
	24		22.530999	11.139698	
	25		23.483051	11.604933	
	26		24.436179	12.070253	
	27		25.390385	12.53566	
	28		26.345671	13.001154	
	29		27.302039	13.466735	
	30		28.25949	13.932402	
	10		9.4864613	4.7211896	
	11		10.440957	5.1937912	
	12		11.396523	5.6664808	
	13		12.353161	6.1392582	
	14		13.310874	6.6121235	
	15		14.269663	7.0850767	
	16		15.22953	7.5581179	
	17		16.190476	8.0312471	
	18		17.152504	8.5044643	
	19		18.115616	8.9777695	
26	20	1.016	19.079812	9.4511628	6.6627796
	21		20.045096	9.9246442	
	22		21.011468	10.398214	
	23		21.978932	10.871871	
	24		22.947487	11.345617	
	25		23.917137	11.819451	
	26		24.887884	12.293373	
	27		25.859729	12.767383	
	28		26.832673	13.241482	
	29		27.80672	13.715669	
	30		28.78187	14.189944	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		9.6545285	4.8048327	
	11		10.625934	5.2858072	
	12		11.59843	5.7668712	
	13		12.572016	6.2480245	
	14		13.546697	6.7292674	
	15		14.522472	7.2105997	
	16		15.499344	7.6920216	
	17		16.477315	8.173533	
	18		17.456387	8.6551339	
	19		18.436562	9.1368245	
27	20	1.034	19.41784	9.6186047	6.6597926
	21		20.400225	10.100474	
	22		21.383719	10.582434	
	23		22.368322	11.064483	
	24		23.354037	11.546622	
	25		24.340866	12.028851	
	26		25.328811	12.511169	
	27		26.317873	12.993577	
	28		27.308055	13.476075	
	29		28.299358	13.958663	
	30		29.291785	14.441341	
	10		9.8244631	4.8894052	
	11		10.812967	5.3788456	
	12		11.80258	5.868377	
	13		12.793303	6.3579994	
	14		13.785139	6.8477129	
	15		14.77809	7.3375174	
	16		15.772157	7.8274131	
	17		16.767342	8.3173998	
	18		17.763647	8.8074777	
	19		18.761074	9.2976467	
28	20	1.0522	19.759624	9.787907	6.6473574
	21		20.759301	10.278258	
	22		21.760105	10.768701	
	23		22.762039	11.259235	
	24		23.765104	11.74986	
	25		24.769303	12.240577	
	26		25.774637	12.731385	
	27		26.781109	13.222284	
	28		27.788719	13.713275	
	29		28.797471	14.204357	
	30		29.807365	14.695531	
	10		9.9813259	4.9674721	
	11		10.985613	5.4647272	
	12		11.991026	5.9620747	
	13		12.997568	6.4595147	
	14		14.005241	6.9570472	
	15		15.014045	7.4546722	
	16		16.023984	7.9523898	
	17		17.035058	8.4501999	
	18		18.047271	8.9481027	
	19		19.060623	9.446098	
29	20	1.069	20.075117	9.944186	6.6524596
	21		21.090755	10.442367	
	22		22.107539	10.94064	
	23		23.12547	11.439006	
	24		24.144551	11.937465	
	25		25.164783	12.436017	
	26		26.186169	12.934661	
	27		27.20871	13.433398	
	28		28.232409	13.932229	
	29		29.257267	14.431152	
	30		30.283286	14.930168	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		10.146592	5.0497212	
	11		11.167507	5.5552096	
	12		12.189568	6.060792	
	13		13.212776	6.5664683	
	14		14.237133	7.0722388	
	15		15.26264	7.5781032	
	16		16.289301	8.0840617	
	17		17.317117	8.5901144	
	18		18.346089	9.0962612	
	19		19.37622	9.6025021	
30	20	1.0867	20.407512	10.108837	6.6374076
	21		21.439966	10.615267	
	22		22.473585	11.12179	
	23		23.508371	11.628408	
	24		24.544325	12.13512	
	25		25.58145	12.641926	
	26		26.619748	13.148827	
	27		27.659219	13.655822	
	28		28.699868	14.162912	
	29		29.741695	14.670096	
	30		30.784703	15.177374	

B.9 RANDOM SMALL

Similar to the previous demand scenario, this scenario models demand as a random distribution; however, this model only permits a maximum of 10 vessels demanded per product variant.

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
10	20	0.8979	16.861972	8.3525581	2.7509549
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	
	10		8.3837535	4.1723978	
	11		9.2272982	4.5900641	
	12		10.071789	5.0078081	
	13		10.917228	5.4256298	
	14		11.763616	5.8435292	
	15		12.610955	6.2615063	
	16		13.459247	6.6795611	
	17		14.308493	7.0976937	
	18		15.158694	7.515904	
	19		16.009854	7.9341922	
20	20	0.8979	16.861972	8.3525581	6.6991098
	21		17.715051	8.771002	
	22		18.569092	9.1895236	
	23		19.424097	9.6081232	
	24		20.280068	10.026801	
	25		21.137006	10.445556	
	26		21.994912	10.864389	
	27		22.85379	11.283301	
	28		23.713639	11.70229	
	29		24.574462	12.121357	
	30		25.436261	12.540503	

V (m^3)	P (MPa)	R (m)	Ts (mm)	Th (mm)	Lo (m)
	10		10.146592	5.0497212	
	11		11.167507	5.5552096	
	12		12.189568	6.060792	
	13		13.212776	6.5664683	
	14		14.237133	7.0722388	
	15		15.26264	7.5781032	
	16		16.289301	8.0840617	
	17		17.317117	8.5901144	
	18		18.346089	9.0962612	
	19		19.37622	9.6025021	
30	20	1.0867	20.407512	10.108837	6.6374076
	21		21.439966	10.615267	
	22		22.473585	11.12179	
	23		23.508371	11.628408	
	24		24.544325	12.13512	
	25		25.58145	12.641926	
	26		26.619748	13.148827	
	27		27.659219	13.655822	
	28		28.699868	14.162912	
	29		29.741695	14.670096	
	30		30.784703	15.177374	

APPENDIX C

JAVA CODE FOR THE HEARING AID EXAMPLE PROBLEM

In this appendix, the computer code for the implementation of the hearing aid example problem (see Chapter 5) is presented. This computer code is programmed in the JAVA programming language. A brief description of each class file and the functions of each file are presented in Appendix C.1. The actual code is presented in Appendix C.2.

C.1 LIST OF CLASS FILES AND FUNCTIONS

C.1.1 List of Class Files

clsArrayOutput

A class analogous to the similarly named class of the pressure vessel example (see Appendix A), the instantiation of this class creates a BufferedWriter in which an output file is written. This output file contains all of the pertinent data for each node of the space of capacity.

clsCapacitySpace

This class is very similar to clsMarketSpace of the pressure vessel example (Appendix A). Given a one-dimensional space of capacity, this class discretizes the space and returns a set of nodes.

clsDecision1

Despite its name, this class is the instantiation of "Decision 0," the selection of the best combination of road width, batch size, and layer thickness for a given capacity.

clsFDMCost

This class calculates the cost of producing a single capacity requirement.

clsFDMQuality

This class calculates the quality of a part given its layer thickness and road width.

clsFDMThroughput

This class calculates the amount of time needed to build a single capacity requirement.

clsFDMUtility

This class is used to calculate the individual utility functions of time, cost, and quality as well as the expected utility.

clsFinalOutput

This class, similar to clsArrayOutput, creates an output pfile to which the results of the best combination of the ranges of modes of managing customization are written.

clsHearingAid

This class is the backbone of the hearing aid example problem computer code. The main solution algorithm is run through this piece of code, and is the file called by iSIGHT in its iterative solution method.

clsMachineInfo

This class stores all of the pertinent data of each machine type.

clsReadFile

This class holds the functions for reading in the input file.

C.1.2 List of Functions

clsArrayOutput

addText: This function adds text to the output file.

close: This function closes the output file.

clsCapacitySpace

getNodes: this method uses the desired discretization resolution and returns the number of nodes in the space.

getNodeArray: this function returns the two-dimensional array which contains the location of each node as well as all of their pertinent information.

getXDim: this function returns the number of nodes that discretize the space of customization.

clsDecision1

getBestCombo: the heart of clsDecision1, this function performs the exhaustive search of process parameters to identify the combination that provides the highest expected utility for the inputted capacity.

getUtility: returns the total expected utility of the most recent combination of process parameters attempted.

clsFDMCost

getTotalCost: returns the total cost of producing a single capacity requirement.

clsFDMQuality

getQuality: returns the quality of the parts for a single capacity requirement

clsFDMThroughput

setMachine: given the name of the machine, this function instantiates a new *clsMachineInfo* in order to get the necessary data for the scan speed of the process

getTotalBuildTime: returns the total build time needed to produce a given capacity

getBuildMaterialLength: returns the amount of build material used to produce a single part

getSupportMaterialLength: returns the amount of support material used for the production of a single part

getTimePerPart: returns the amount of time needed to build a single part

getTimePerBatch: returns the amount of time needed to build a single batch of parts

clsFDMUtility

evalTimeUtility: returns the value of the time utility function.

evalCostUtility: returns the value of the cost utility function

evalQualityUtility: returns the value of the quality utility function

evalExpectedUtility: returns the value of the expected utility function.

clsFinalOutput

addText: adds text to the output file

close: closes the output file

clsHearingAid

factorCheck: this function divides one number into another to test if they are multiples/factors of one another.

clsMachineInfo

getInfo: this function returns an array containing the data needed for analysis for the specifc machine listed in the input file.

clsReadFile

getNMachines: this function returns the number of machines, as read from the input file

getMachineType: this function returns the machine type, as read from the input file

getDBS: this function returns the range of the Mode C1, as read from the input file

getDLT: this function returns the range of the Mode C2, as read from the input file

getDRW: this function returns the range of the Mode C3, as read from the input file

C.2 SOFTWARE CODE

clsArrayOutput

```
/*
 * clsArrayOutput.java
 */

 /* -- NOTES --
  * This class creates a BufferedWriter
  * it adds text with the addText() function
  * which requires an input of the nodeArray
  * it outputs a text file to be analyzed with Excel
 */

 import java.io.*;

 public class clsArrayOutput
 {
      BufferedWriter out;
      //---- Creates new clsArrayOutput ----//
      public clsArrayOutput(double res)
      {
            try
            {
                  out = new BufferedWriter(new FileWriter("NodeInfo_" +
                       res + ".txt"));
                  out.write("Resolution = " + res);
                  out.write("\n");
            }
            catch (IOException e)
            {
                  System.err.println("FileStream error " + e);
            }
      }//end of clsArrayOutput constructor
```

```
//----FUNCTIONS----//

//FUNCTION: addText//
//function adds text to the opened file
public void addText(String MachineType, double MachineN, double
     dBS, double dRW, double dLT, double[][] nodeArray, int
     arraylength)
{
      try
      {

out.write("+++++++++++++++++++++++++++++++++++++++++++++");
            out.write("\n");
            out.write(MachineN + "    " + MachineType + "\n");
            Double ddBS = new Double(dBS);
            String sdBS = ddBS.toString();
            out.write("dBS: ");
            out.write(sdBS);
            out.write("   ");
            Double ddRW = new Double(dRW);
            String sdRW = ddRW.toString();
            out.write("dRW: ");
            out.write(sdRW);
            out.write("   ");
            Double ddLT = new Double(dLT);
            String sdLT = ddLT.toString();
            out.write("dLT: ");
            out.write(sdLT);
            out.write("\n");

            out.write("Node#   XLoc   BatchSize    RoadWidth
               LayerThickness  Cost    Time   Quality
               ExpectedUtility   Constraint");
            out.write("\n");

            for (int i=0; i<arraylength; i++)
            {
                  Integer k = new Integer(i);
                  String r = k.toString();
                  out.write(r);
                  out.write("        ");

                  for (int j= 0; j<9; j++)
                  {
                        Double d = new Double(nodeArray[i][j]);
                        String s = d.toString();
                        out.write(s);
                        out.write("     ");
                  }
                  out.write("\n");
            }
      }
      catch (IOException e)
      {
            System.err.println("FileStream error " + e);
```

```
            }
      }//end addText()

      //FUNCTION: close//
      //function closes file
      public void close()
      {
            try
            {
                  out.close();
            }
            catch (IOException e)
            {
                  System.err.println("FileStream error " + e);
            }
      }//end close()

}//end of clsArrayOutput
```

clsCapacitySpace

```
/*
 * clsCapacitySpace.java
 */

 /* -- NOTES --
  * This class requires an input of
  * the dimensions of a 1D market space
  * and a resolution number.
  * The output is an array of nodes.
 */

 import java.lang.Number.*;

 public class clsCapacitySpace
 {

      //---- Constants ----//
      double res; //resolution variable
      double xo; //initial x value
      double dimx; //length of x axis (continuous)

      int xint = 0; //dimension of x-axis array

      double nodearray[][]; //a 2D array that stores all info about a
                              node

      //---- Creates new clsCapacitySpace ----//
      public clsCapacitySpace(double dres, double dxo, double ddimx)
      {
            //---- Problem Dependent Variables ----//
            res = dres;
            xo = dxo;
            dimx = ddimx;

            //---- Analysis ----//
```

```
        //make array of dim(xint), start with xo, add res, end with
           xn
        getNodes();  //establishes number of nodes
        //create nodearray[]
        //# of rows = # of nodes (# of X)
        //# of columns = 11 (xlocation, batch size, road width,
             layer thickness, cost, time, quality, expectedutility,
             constraint)
        nodearray = new double[xint][9];
        nodearray[0][0] = xo;

        //fill nodearray with x locations
        for (int i=1; i<xint; i++)
        {
              nodearray[i][0] = nodearray[i-1][0] + res;
        }

        //just in case the res is not a factor of the dim...
        nodearray[xint-1][0] = xo + dimx;

    } //end constructor: clsMarketSpace()


    //------- FUNCTIONS -------//

    //---- FUNCTION: getNodes() ----//
    public void getNodes()
    {
        //used in the constructor to establish the number of nodes
        Double x = new Double (dimx / res);

        xint = x.intValue() + 1;

        double xd = xint+0.0;
        double decimaltestx = (dimx/res) - xd;
        if (decimaltestx > 0)
        {
              xint = xint++;
              System.out.println("ERROR: Resolution is not a factor
                   of Dimension!");
        }
    }//end getNodes

    //---- FUNCTION: getNodeArray() ----//
    public double[][] getNodeArray()
    {
        return nodearray;
    } //end of getNodeArray


    //---- FUNCTION: getXDim() ----//
    public int getXDim()
    {
        return xint;
    }//end getXDim

//end of FUNCTIONS
```

```
}//end clsMarketSpace
```

clsDecision1

```
/*
 * clsDecision1.java
 */

 /* -- NOTES --
      //given a capacity, iterate through BS, LT, and RW
      //to find combo that gives highest utility
 */

 public class clsDecision1
 {

      double ExpectedUtility;
      //need place to store best combo of iterations
      //bestcombo[batchsize, roadwidth, layerthickness,
expectedutility]
      double bestcombo[] = new double[8];

      public clsDecision1()
      {
      }//end of clsDecision1 constructor

      //==== FUNCTIONS ====//

      public double[] getBestCombo(String dmachinetype, double
           dmachinenum, double dcapacity, double batchsize, double
           roadwidth, double layerthickness)
   {
            double capacity = dcapacity;
            String MachineType = dmachinetype;
            double machinenum = dmachinenum;
            double BS = batchsize;
            double RW = roadwidth;
            double LT = layerthickness;

            double setuppenalty = 0;
            if (RW < 0 || LT < 0)
            {
                  setuppenalty = 1;
            }

            clsMachineInfo MI = new clsMachineInfo();
            double machineInfo[] = MI.getInfo(MachineType);

            double lowBS = 20; //lower bounds for batch size is 20
                  parts
            double upBS = machineInfo[5]; //upper bounds for batch size
                  is max parts of specifc machine
```

```
double lowRW = machineInfo[2]; //lower bounds for RW for
      specific machine
double upRW = machineInfo[3]; //upper bounds for RW for
      specific machine
double lowLT = machineInfo[0]; //lower bounds for LT for
      specific machine
double upLT = machineInfo[1]; //upper bounds for LT for
      specific machine

double BSres = (upBS - lowBS) / 100;
double LTres = (upLT - lowLT) / 25; //resolution of
    exhaustive search is 25 nodes
double RWres = (upRW - lowRW) / 25;

double BSit;
double LTit;
double RWit;

bestcombo[6] = 0;

BSit = lowBS;
while (BSit < upBS)
{
      if (BS > 0) //if BS is already known, use it!
      {BSit = BS;}

      RWit = lowRW;
      while (RWit < upRW)
      {
            if (RW > 0) //if RW is already known, use it!
            {RWit = RW;}

            LTit = lowLT;
            while (LTit < upLT)
            {
            if (LT > 0) //if LT is already known, use it!
            {LTit = LT;}

            //evaluate number of batches
            double Nbatch = 7*(capacity / BSit); //number
                  of batches needed to meet demand
                  requirement

            //evaluate wait time
            double waittime = (BSit * 24 * 3600) /
                  capacity; //total time to wait for
                  batchsize to accumulate (s)

            //evaluate total time for capacity
            clsFDMThroughput timeEval = new
                  clsFDMThroughput(MachineType, LTit, RWit,
                  BSit, Nbatch);
            double timeperpart = timeEval.getTimePerPart();
                  //time to build one part (s)
            double timeperbatch =
                  timeEval.getTimePerBatch(); //time to
                  build one batch (s)
```

```
double totaltime =
      timeEval.getTotalBuildTime(); //time to
      build whole capacity (s)
double newtotaltime = (totaltime +
      (setuppenalty*30*60))/machinenum;
      //include extra setup time of 30min. &
      number of machines
//calculate cost for capacity
double Lbuild =
      timeEval.getBuildMaterialLength();
      //amount of material for one part
double Lsupp =
      timeEval.getSupportMaterialLength();
      //amount of material for one part
clsFDMCost costEval = new
      clsFDMCost(machinenum, machineInfo[6],
      capacity, Nbatch, LTit, RWit, BSit,
      Lbuild, Lsupp, totaltime);
double totalcost = costEval.getTotalCost();
totalcost = totalcost + (setuppenalty*50);
      //include extra setup cost of $50
//not concerned with machine cost and
      maintenance cost until next decision

//evaluate quality
clsFDMQuality qualityEval = new
      clsFDMQuality(LTit, RWit);
double quality = qualityEval.getQuality();

//evaluate utility
double kt = 0.25;
double kc = 0.25;
double kq = 0.5;

clsFDMUtility utilityEval = new
      clsFDMUtility();
double timeutility =
      utilityEval.evalTimeUtility(newtotaltime,
      kt);
double costutility =
      utilityEval.evalCostUtility(totalcost,
      kc);
double qualityutility =
      utilityEval.evalQualityUtility(quality,
      kq);
ExpectedUtility =
      utilityEval.evalExpectedUtility();

if (timeutility == 0 || costutility == 0 ||
      qualityutility == 0)
{
ExpectedUtility = 0;
}
double constraint = (newtotaltime + waittime) ;

if (constraint > 7*24*3600) //if constraint >
      week (* numMachines?)
```

```
                        {
                        ExpectedUtility = 0;
                        }

                        if (bestcombo[6] < ExpectedUtility)
                        {
                        //bestcombo[batchsize, layerthickness,
                              roadwidth, expectedutility, constraint]
                        bestcombo[0] = BSit;
                        bestcombo[1] = RWit;
                        bestcombo[2] = LTit;
                        bestcombo[3] = totalcost;
                        bestcombo[4] = newtotaltime;
                        bestcombo[5] = quality;
                        bestcombo[6] = ExpectedUtility;
                        bestcombo[7] = constraint;
                        }

                        //increase iteration step
                        LTit = LTit + LTres;
                        if (LT > 0) //if LT is known we don't have to
                              run it again
                        {LTit = upLT;}
                        }//end of LT while

                        //increase iteration step
                        RWit = RWit + RWres;
                        if (RW > 0) //if LT is known we don't have to
                             run it again
                        {RWit = upRW;}
                  } //end of RW while

                  //increase iteration step
                  BSit = BSit + BSres;
                  if (BS > 0) //if BS is know we don't have to run it
                       again
                  {BSit = upBS;}
            } //end of BS while

      return bestcombo;

      }//end of getBestCombo()

      public double getUtility()
   {
      return ExpectedUtility;
   }//end of getUtility

} //end of clsDecision1
```

clsFDMCost

```
/*
 * clsFDMCost.java
 */
```

```
/* -- NOTES --
 * This class requires an input of
 * batchsize, LT, RW, machine type, machine #
 * The output is amount of time to build *one batch*
*/

public class clsFDMCost
{

     //---- Global Variables----//
     double Costtotal;

     //---- Creates new clsFDMCost ----//
     public clsFDMCost(double dmachinenum, double dmachinecost, double
           dcapacity, double dNbatch, double LT, double RW, double BS,
           double Lbuild, double Lsupp, double time)
     {

           double machinenum = dmachinenum; //number of machines
           double machinecost = dmachinecost; //cost of one machine
           double capacity = dcapacity; //demand per day
           double Nbatch = dNbatch; //number of batches required to
                 meet capacity at current batchsize
           double layerthickness = LT; //current LT for capacity
           double roadwidth = RW; //current RW for capacity
           double batchsize = BS; //current BS for capacity

           double buildlength = Lbuild; //length of build material for
                  ONE part
           double supportlength = Lsupp; //length of support material
              for ONE part

           double buildtime = time; //build time per batch, s

           //---- Constants ----//
           double ABSmaterialcost = 4.31 / 16387.064; //cost for ABS,
              $/mm3
           double ELASTmaterialcost = 5.52 / 16387.064;
           double supportcost = 3.79 / 16387.064; //cost for
                waterworks, $/mm3
           double laborcost = 25; //25 $/hr
           double operationcost = 80; //80 $/hr
           double maintcost = 50000; //50,000 $/year

           //---- ANALYSIS----//
           //calculate volume of material, for one part (~229 m3)
           double materialarea = layerthickness * roadwidth;
           double volbuild = materialarea * buildlength; //volume of
                  build material used, mm3
           double volsupport = materialarea * supportlength; //volume
of support material used, mm3

           //calculate material cost for entire capacity
           double buildmatcost = volbuild * ABSmaterialcost; //cost of
                build material for one part, $
           double suppmatcost = volsupport * supportcost; //cost of
                support material for one part, $
```

```
            double Costmaterial = buildmatcost + suppmatcost; //cost of
                material for ONE part

            Costmaterial = Costmaterial * batchsize * Nbatch; //cost of
                material for entire capacity need

            //calculate labor cost for entire capacity
            // techhrcost * ophours
            double Costlabor = laborcost * (buildtime / 3600);

            //calculate operation cost for entire capacity
            //machhrcost * ophours
            double Costoperation = operationcost * (buildtime/3600);

            //calculate total cost of capacity requirement
            Costtotal = Costmaterial + Costlabor + Costoperation; //($
                for capacity) (w/o maintenance and machine cost)

            //calculate machine and maintenance costs
            double costmaintenance = 50000 * machinenum;
            double costmachine = machinenum * machinecost;
            double costMMpercapacity = (7*(costmaintenance +
                costmachine))/365;

            //total cost
            Costtotal = Costtotal + costMMpercapacity;

        } //end of clsFDMCost constructor


    //==== FUNCTIONS ====//

     public double getTotalCost()
     {
       return Costtotal;
     }//end of getTotalCost

     //== End of FUNCTIONS ==//

  } //end of clsFDMCost
```

clsFDMQuality

```
/*
 * clsFDMQuality.java
 */

 /* -- NOTES --
  * This class requires an input of
  * LT, RW
  * The output is quality for specific part
 */

 public class clsFDMQuality
```

```
{

      //---- Global Variables----//
      double LT; //layer thickness (mm)
      double RW; //road width (mm)
      double Quality; //quality metric

      //---- Creates new clsFDMQuality ----//
      public clsFDMQuality(double layerthickness, double roadwidth)
      {
            LT = layerthickness;
            RW = roadwidth;

            Quality = LT + RW;

      }//end of clsFDMQuality constructor

      //==== FUNCTIONS ====//
      //--- Function: getQuality() ---//
    public double getQuality()
    {
        return Quality;
    }//end getQuality()
    //== End of FUNCTIONS ==//

}//end of clsFDMQuality
```

clsFDMThroughput

```
/*
 * clsFDMThroughput.java
 */

 /* -- NOTES --
  * This class requires an input of
  * batchsize, LT, RW,and machine type
  * The output is total amount of time to build capacity requirement
  * (s)
 */

 public class clsFDMThroughput
 {
      //---- Constants ----//
      //Dimensions of the hearing aid... in (mm)
   double l;                //Length of a hearing aid
   double w;                //Width of a hearing aid
   double h;                //Height of a hearing aid
   double ts;               //Thickness of the shell

   //Dimensions of the platform (in mm)
   double L;                //Length
   double W;                //Width
   double H;                //Height

   //Scan velocities...
```

```
double Vsb;              //Scan speed at the boundary
double Vsf;              //Speed at which the interiors are filled...
double Vss;              //Scan Speed for supports

//Road widths
double Rw;              //Road width

//Layer thickness
double tl;               //Layer thickness

//Batch Size
double batchsize;
double Nbatch; //number of batches

//Total time for build
double Ttotal; //total time to build capacity requirement (s)

//Total time for part
double timeperpart; //total time to build one part (s)

//Total time for batch
double timeperbatch; //(s)

   //Material usage
   double buildmaterial;   //build material length (mm)
   double supportmaterial; //support material length (mm)

   //---- Creates new clsFDMThroughput ----//
   public clsFDMThroughput(String dMachineType, double
        dlayerthickness, double droadwidth, double dbatchsize, double
        dNbatch)
   {
   //begin by setting machine parameters (L,W,H & Scan Velocities)
         String MachineType = dMachineType;
         setMachine(MachineType);
         //establish input
         tl = dlayerthickness;
         Rw = droadwidth;
         batchsize = dbatchsize;
         Nbatch = dNbatch; //number of batches

         //next, set part dimensions
         w = (10.72 + 6.36)/ 2; //mm
         l = (18.7 + 10.5) / 2; //mm
         h = 21.78; //mm
         ts = 1.11; //mm

         //calculate set up time (one per batch)
         double Tsetup = 30 * 60; //setup time; 30 min (s)

         //calculate number of layers
         //number of layers is calculated by height of part divided
             by layerthickness
         double Nl = h / tl;

         //calculate time to build each layer
```

```
        //time to lower nozzle + time to draw boundaries + time to
            fill + time to draw support
        double Tlower = 5.0/batchsize; //time to lower nozzle, (s)
        double Tboundary = (2* (l+w)/Vsb); //time to build a part
           boundary (s)

        double Tfill;//time to fill boundaries
     if((ts-2*Rw)>0)
  {
        Tfill = ((ts-(2*Rw))*(l+w)/(Rw*Vsf)); //(s)
  }
  else
  {
        Tfill = 0.0; //(s)
        }

        double Tsupp = 0.25 * Tfill; //time to draw supports; 25%
           of fill, (s)

        //time to build a single layer of a part
        double Tl = Tlower + Tboundary + Tfill + Tsupp;

        //calculate time to build base of a part
        double Tbase = 4*(l*w)/(Rw*Vss);

        //Calculate total time
        //setup time + (# layers * time per layer) + time for base
        timeperpart = (Tl*Nl) + Tbase;
        timeperbatch = Tsetup + (batchsize * timeperpart);
        //seconds
        Ttotal = timeperbatch * Nbatch;


        //calculate material usage
  //The length of material used for one layer is equal to the
     velocity multiplied by time
  double L_boundary = Vsb * Tboundary;   //For one layer
  double L_Interior = Vsf * Tfill;
  double L_Supports = Vss * Tsupp;
  //This is the length of material (in mm) required per part
        buildmaterial = (L_boundary + L_Interior)* Nl;
  //In this, we have also added the material required to build
        bottom 4 layers
     supportmaterial = (L_Supports * Nl) + (4*l*w/Rw);


  } //end of clsFDMThroughput constructor

//==== FUNCTIONS ====//
public void setMachine(String sMachine)
{
        clsMachineInfo MI = new clsMachineInfo();
     double machineinfo[] = MI.getInfo(sMachine);

   if (sMachine.equals("ProdigyPlus"))
   {
       L = 203;
```

```
            W = 203;
            H = 305;

            Vsb = machineinfo[4];
            Vsf = Vsb;
            Vss = Vsb;
        }
        if (sMachine.equals("Maxum"))
        {
            L = 600;
            W = 500;            H = 600;

            Vsb = machineinfo[4];
            Vsf = Vsb;
            Vss = Vsb;

        }
        if (sMachine.equals("Titan"))
        {
            L = 355;
            W = 406;
            H = 406;

            Vsb = machineinfo[4];
            Vsf = Vsb;
            Vss = Vsb;
        }
    }//end of setMachine

    public double getTotalBuildTime()
    {
      return Ttotal; //total build time for capacity (s)
    }//end of getTotalBuildTime

    public double getBuildMaterialLength()
    {
        return buildmaterial;
    }//end getBuildMaterialUsage

    public double getSupportMaterialLength()
    {
            return supportmaterial;
    }//end of getSupportMaterialUsage()

    public double getTimePerPart()
    {
            return timeperpart;
    }//end of getTimePerPart()
    public double getTimePerBatch()
    {
      return timeperbatch;
       }//end of getTimePerBatch

    //=== End of FUNCTIONS ===//


} //end of clsFDMThroughput
```

clsFDMUtility

```
/*
 * clsFDMUtility.java
 */

 /* -- NOTES --
      //evaluate utility of time, cost, and quality
      //evaluate expected utility
 */

 public class clsFDMUtility
 {

      //--- GLOBAL VARIABLES ---//
      double timeutility;
      double costutility;
      double qualityutility;

      double kt;
      double kc;
      double kq;

      double ExpectedUtility;

      //--- CONSTRUCTOR ---//
      public clsFDMUtility()
      {
      }//end of clsFDMUtility constructor

      //==== FUNCTIONS ====//

    public double evalTimeUtility(double dtime, double ktime)
    {
            //convert time to hrs
            double time = dtime / 3600;
            double tmax = 168; //utility = 0
            double tmin = 13; //utility = 1
            timeutility = 2 - java.lang.Math.exp((0.685*(time-
               tmin))/(tmax-tmin));
            kt = ktime;

            return timeutility;
    }//end evalTimeUtility

    public double evalCostUtility(double cost, double kcost)
    {
      double cmax = 25000; //utility = 0
      double cmin = 1000; //utility = 1
      costutility = 2 - java.lang.Math.exp((0.685*(cost-cmin))/(cmax-
           cmin));
      if (costutility < 0) { costutility = 0.01; }
      kc = kcost;
      return costutility;
    }//end evalCostUtility
```

```
    public double evalQualityUtility(double quality, double kquality)
    {
      double qmax = 1.3;
      double qmin = 0.37;
      qualityutility = 2.04 - java.lang.Math.exp(0.6861*(quality-
         qmin)/(qmax-qmin));
      kq = kquality;
      return qualityutility;
    }//end evalQualityUtility

    public double evalCostUtility2(double cost, double kcost)
    {
      double cmax = 25000; //utility = 0
      double cmin = 6000; //utility = 1
      costutility = 2 - java.lang.Math.exp((0.685*(cost-cmin))/(cmax-
         cmin));
      if (costutility < 0) { costutility = 0.01; }
      kc = kcost;
      return costutility;
    }//end evalCostUtility


    public double evalExpectedUtility()
    {
      ExpectedUtility = (kt * timeutility) + (kc * costutility) + (kq *
         qualityutility);
      if (timeutility < 0 || costutility < 0 || qualityutility < 0)
      {
            ExpectedUtility = 0;
      }
      return ExpectedUtility;
    }//end evalExpectedUtility

} //end of clsFDMUtility
```

clsFinalOutput

```
/*
 * clsFinalOutput.java
 */

 /* -- NOTES --
  * This class creates a BufferedWriter
  * it adds text with the addText() function
  * which requires an input of the expected utility
  * it outputs a text file to be analyzed with iSight
 */

 import java.io.*;

 public class clsFinalOutput
 {
      BufferedWriter out;
      //---- Creates new clsFinalOutput ----//
```

```
        public clsFinalOutput(double res)
        {
                try
                {
                        out = new BufferedWriter(new
                             FileWriter("output.txt"));
                }
                catch (IOException e)
                {
                        System.err.println("FileStream error " + e);
                }
        }//end of clsFinalOutput constructor


        //----FUNCTIONS----//
        public void addText(String describe, double value)
        {
                try
                {
                        out.write(describe + "\n");
                        out.write(value + "\n");
                }
                catch (IOException e)
                {
                        System.err.println("FileStream error " + e);
                }
        }

        public void close()
        {
                try
                {
                        out.close();
                }
                catch (IOException e)
                {
                        System.err.println("FileStream error " + e);
                }
        }
 }//end of clsArrayOutput
```

clsHearingAid

```
/*
 * clsHearingAid.java
 */

public class clsHearingAid {

        public static void main (String[] args)
        {
                new clsHearingAid();
        }

        //---- Global Variables ----//
```

```
double pi = java.lang.Math.PI; //pi, 3.14...

double dBS = 0.0; //range of commonality of batch size;
    D(parts/batch)
double dLT = 0.0; //range of commonality of layer thickness;
    D(demand/day)
double dRW = 0.0; //range of commonality of road width;
    D(demand/day)
String MachineType = null; //name of machine
double MachineN = 0; //number of machines

double nodeArray[][];
double machineInfo[];

double batchsize;

//---- Creates new clsHearingAid ----//
public clsHearingAid()
{
      //---- First, define and discretize space ----//
      double res = 5.0; //resolution of discretization

      double xo = 120.0; //initial point of x-dimension of space,
         min capacity
      double xn = 1000.0; //end point of x-dimension of space,
         max capacity
      double dimx = xn - xo; //length of x axis

      //ues to find the largest expected utility; the major
         output
      double expectedutility = 0.0;

      //ALGORITHM DESCRIPTION
      // want to use this as an executable for isight
      // STEP 1) read input file which has DRW, DLT, DBS, Machine
         type and machine #
      // STEP 2) discretize space
      // STEP 3) discretize space for DRW, DLT, and DBS
      // STEP 4) decide which nodes in Step 2 have common RW, LT,
         and BS
      // STEP 5) iterate among RW, LT, and BS (clsDecision1) to
         decide which values should be made common
      // STEP 6) assign common values to appropriate nodes
      // STEP 7) analyze time, cost and quality at each point
      // STEP 8) take average of each across space, calculate
         expected utility
      // STEP 9) spit out expected utility to an output file
         along with node info
      // IT'S THAT SIMPLE! No tedious exhaustive search of DRW,
         DBS, DLT! Hooray!

      //----- STEP 1 -----//
      // STEP 1) read input file which has DLT, DRW, Machine type
         and machine #
      clsReadFile ReadFile = new clsReadFile();
      MachineN = ReadFile.getNMachines();
      int machinename = ReadFile.getMachineType();
```

```
if (machinename == 1) {MachineType = "ProdigyPlus";}
if (machinename == 2) {MachineType = "Titan";}
if (machinename == 3) {MachineType = "Maxum";}
dBS = ReadFile.getDBS();
dRW = ReadFile.getDRW();
dLT = ReadFile.getDLT();

//determine machine type from input file
clsMachineInfo MI = new clsMachineInfo();
machineInfo = MI.getInfo(MachineType);

//----- STEP 2 -----//
// STEP 2) discretize space
//Use CapacitySpace class to define Market
//get the node array ... a place to store each node's info
clsCapacitySpace CapacitySpace = new clsCapacitySpace(res,
   xo, dimx);
nodeArray = CapacitySpace.getNodeArray();
int numnodes = CapacitySpace.getXDim();

//establish output file
clsArrayOutput ArrayOutput = new clsArrayOutput(res);

//----- STEP 3 -----//
// STEP 3) discretize space for DLT, DRW, and DBS
//IMPORTANT: BSnodes < RWnodes < LTnodes as per hierarchy!!

//establish set of nodes for BS
Double dBSnodes = new Double(dimx/dBS);
int BSnodes = dBSnodes.intValue()+1;
int z = factorCheck(dimx, dBS);
BSnodes = BSnodes + z;
double BSnodearray[] = new double[BSnodes];
BSnodearray[0] = xo;
for (int i=1; i<BSnodes; i++)
{
      BSnodearray[i] = BSnodearray[i-1] + dBS;
}
if (z>0) { BSnodearray[BSnodes-1] = xn; }

//establish set of nodes for RW
Double dRWnodes = new Double(dimx/dRW);
int RWnodes = dRWnodes.intValue()+1;
int y = factorCheck(dimx, dRW);
RWnodes = RWnodes + y;
double RWnodearray[] = new double[RWnodes];
RWnodearray[0] = xo;
for (int i=1; i<RWnodes; i++)
{
      RWnodearray[i] = RWnodearray[i-1] + dRW;
}
if (y>0) { RWnodearray[RWnodes-1] = xn; }

//establish set of nodes for LT
Double dLTnodes = new Double(dimx/dLT);
int LTnodes = dLTnodes.intValue() + 1;
int x = factorCheck(dimx, dLT);
```

```
LTnodes = LTnodes + x;
double LTnodearray[] = new double[LTnodes];
LTnodearray[0] = xo;
for (int i=1; i<LTnodes; i++)
{
      LTnodearray[i] = LTnodearray[i-1] + dLT;
}
if (x>0) { LTnodearray[LTnodes-1] = xn; }

//----- STEPS 4, 5, 6 -----//
// STEP 4) decide which nodes in Step 2 have common RW, LT,
     and BS
//we'll be working backwards
// STEP 5) iterate among RW, LT, and BS (clsDecision1) to
   decide which values should be made common
// STEP 6) assign common values to appropriate nodes
// STEP 7) analyze time, cost and quality at each point

//for last node first
int n = numnodes - 1; //counter for nodeArray

clsDecision1 Decision1 = new clsDecision1();

//do last node first...
nodeArray[n][1] = nodeArray[n][2] = nodeArray[n][3] = -
   9999;
double bestcombo[] = Decision1.getBestCombo(MachineType,
   MachineN, nodeArray[n][0], nodeArray[n][1],
   nodeArray[n][2], nodeArray[n][3]);
nodeArray[n][1] = bestcombo[0];
nodeArray[n][2] = bestcombo[1];
nodeArray[n][3] = bestcombo[2];
nodeArray[n][4] = bestcombo[3];
nodeArray[n][5] = bestcombo[4];
nodeArray[n][6] = bestcombo[5];
nodeArray[n][7] = bestcombo[6];
nodeArray[n][8] = bestcombo[7];

int b = BSnodes - 2; //counter for BSnodearray (Batchsize)
int r = RWnodes - 2; //counter for RWnodearray (road width)
int l = LTnodes - 2; //counter for LTnodearray (layer
   thickness)


double bscap; //capacity cutoff of current BS element
bscap = BSnodearray[b];
double rwcap; //capacity cutoff of current BS element
rwcap = RWnodearray[r];
double ltcap; //capacity cutoff of current LT element
ltcap = LTnodearray[l];


for (int i=n-1; i>0; i--)
{
      if (nodeArray[i][0] > bscap)
      {
            nodeArray[i][1] = nodeArray[i+1][1];
```

```
        }
        if(nodeArray[i][0] <= bscap)
        {
              b = b-1;
              bscap = BSnodearray[b];
              nodeArray[i][1] = -nodeArray[i][0];
        }

        if (nodeArray[i][0] > rwcap)
        {
              nodeArray[i][2] = nodeArray[i+1][2];
        }
        if(nodeArray[i][0] <= rwcap)
        {
              r = r-1;
              rwcap = RWnodearray[r];
              nodeArray[i][2] = -nodeArray[i][0];
        }

        if (nodeArray[i][0] > ltcap)
        {
              nodeArray[i][3] = nodeArray[i+1][3];
        }
        if(nodeArray[i][0] <= ltcap)
        {
              l = l-1;
              ltcap = LTnodearray[l];
              nodeArray[i][3] = -nodeArray[i][0];
        }

        //analysis of node, given the LT, RW, and BS...
        bestcombo = Decision1.getBestCombo(MachineType,
             MachineN, nodeArray[i][0], nodeArray[i][1],
             nodeArray[i][2], nodeArray[i][3]);
        nodeArray[i][1] = bestcombo[0]; ///changes them from
             -capacity
        nodeArray[i][2] = bestcombo[1];
        nodeArray[i][3] = bestcombo[2];
        nodeArray[i][4] = bestcombo[3];
        nodeArray[i][5] = bestcombo[4];
        nodeArray[i][6] = bestcombo[5];
        nodeArray[i][7] = bestcombo[6];
        nodeArray[i][8] = bestcombo[7];

  } //end of for stepping through nodeArray

  //don't forget for first node
  nodeArray[0][1] = nodeArray[1][1];
  nodeArray[0][2] = nodeArray[1][2];
  nodeArray[0][3] = nodeArray[1][3];

  bestcombo = Decision1.getBestCombo(MachineType, MachineN,
        nodeArray[0][0], nodeArray[0][1], nodeArray[0][2],
        nodeArray[0][3]);
  nodeArray[0][1] = bestcombo[0];
  nodeArray[0][2] = bestcombo[1];
  nodeArray[0][3] = bestcombo[2];
```

```
nodeArray[0][4] = bestcombo[3];
nodeArray[0][5] = bestcombo[4];
nodeArray[0][6] = bestcombo[5];
nodeArray[0][7] = nodeArray[1][7];
nodeArray[0][8] = bestcombo[7];

//create output file
ArrayOutput.addText(MachineType, MachineN, dBS, dRW, dLT,
    nodeArray, numnodes);
ArrayOutput.close();

//----- STEP 8 -----//
// STEP 8) take average of each across space, calculate
   expected utility
//calculate total time to build all capacity

//calculate number of infeasible solns
double infeasible = 0;
for (int i=0; i < numnodes; i++)
{
      if (nodeArray[i][7] == 0) {infeasible++;}
}

double totaltime = 0;
for (int i=0; i < numnodes; i++)
{
      totaltime = totaltime + nodeArray[i][5];
}
double avgtime = totaltime / (numnodes - infeasible);

//calculate cost to build all points at each capacity
double totalcost = 0;
for (int i=0; i < numnodes; i++)
{
      totalcost = totalcost + nodeArray[i][4];
}
double avgcost = totalcost / (numnodes-infeasible);

//calc average quality of parts
double totalquality = 0;
for (int i=0; i < numnodes; i++)
{
      totalquality = totalquality + nodeArray[i][6];
}
double avgquality = totalquality / (numnodes-infeasible);

double kt = 0.25;
double kc = 0.25;
double kq = 0.5;

clsFDMUtility OverallUtility = new clsFDMUtility();
double timeutility =
    OverallUtility.evalTimeUtility(avgtime, kt);
double costutility =
    OverallUtility.evalCostUtility(avgcost, kc);
double qualityutility =
    OverallUtility.evalQualityUtility(avgquality, kq);
```

```
            double FinalUtility = OverallUtility.evalExpectedUtility();

            double deviationplus = FinalUtility - 1.0; //over-achieving
            double deviationneg = 1.0 - FinalUtility; //under-achieving
            if (deviationplus < 0) {deviationplus = 0;}
            if (deviationneg < 0) {deviationneg = 0;}

            //----- STEP 9 -----//
            // STEP 9) spit out expected utility to an output file
                along with node info
            clsFinalOutput FinalOutput = new clsFinalOutput(res);
            FinalOutput.addText("AVGtime", avgtime);
            FinalOutput.addText("Utime", timeutility);
            FinalOutput.addText("AVGcost", avgcost);
            FinalOutput.addText("Ucost", costutility);
            FinalOutput.addText("AVGquality", avgquality);
            FinalOutput.addText("Uquality", qualityutility);
            FinalOutput.addText("E[U(D)]", FinalUtility);
            FinalOutput.addText("d+", deviationplus);
            FinalOutput.addText("d-", deviationneg);
            FinalOutput.close();

      } //end of clsHearingAid() constructor


      //---- FUNCTIONS ----//

      //FUNCTION: factorCheck()//
      public int factorCheck(double n, double m)
      {
            Double x = new Double(n/m);
            int y = x.intValue();
            double z = y+0.0;
            double fractioncheck = (n/m) - z;
            //System.out.println("fractioncheck = " + fractioncheck);
            int flag = 0;
            if (fractioncheck < 0 || fractioncheck > 0)
            {
                  flag = 1;
            }
            return flag;
      }//end of factorCheck()

      //end of FUNCTIONS

} //end of clsHearingAid
```

clsMachineInfo

```
/*
 * clsMachineInfo.java
 */

 /* -- NOTES --
  * This class requires provides info on each machine type
```

```
 * the output is an array of machine specs
 * capacity, cost, scan speed
*/

public class clsMachineInfo
{

     //---- Creates new clsMachineInfo ----//
     public clsMachineInfo()
     {
     } //end of clsMachineInfo constructor


     //---- FUNCTIONS ----//

     //FUNCTION: getInfo()//
     public double[] getInfo(String dMachineType)
     {

           String MachineType = dMachineType;

           //create array to store machineinfo in
           //LBLT, UBLT, LBRW, UBRW, ScanSpeed, Max Parts, Cost
           double machineinfo[] = new double[7];

           if (MachineType.equals("ProdigyPlus"))
           {
           //Based on the platform size of: 203*203*305
           machineinfo[0] = 0.178; //mm
           machineinfo[1] = 0.33; //mm
           machineinfo[2] = 0.19; //mm
           machineinfo[3] = 0.21; //mm
           machineinfo[4] = 64; //mm/s
           machineinfo[5] = 234; //parts
           machineinfo[6] = 70000; //$
           }

           if (MachineType.equals("Titan"))
           {
           //Based on the platform size of: 355*406*406
           machineinfo[0] = 0.24; //mm
           machineinfo[1] = 0.26; //mm
           machineinfo[2] = 0.19; //mm
           machineinfo[3] = 0.21; //mm
           machineinfo[4] = 127; //mm/s
           machineinfo[5] = 818; //parts
           machineinfo[6] = 210000; //$
           }

           if (MachineType.equals("Maxum"))
           {
           //Based on the platform size of: 600*500*600
           machineinfo[0] = 0.127; //mm
           machineinfo[1] = 0.25; //mm
           machineinfo[2] = 0.193; //mm
           machineinfo[3] = 0.965; //mm
           machineinfo[4] = 254; //mm/s
```

```
            machineinfo[5] = 1703; //parts
            machineinfo[6] = 260000; //$
            }

            return machineinfo;
      }//end of getInfo()

      //end of FUNCTIONS

 } //end of clsMachineInfo
```

clsReadFile

```
/*
 * clsReadFile.java
 */

 /* -- NOTES --
  * This class reads the input file
  * NOTE: File must be exactly as prescribed
  * # of machines
  * machine type (ProdigyPlus, Maxum, or Titan)
  * Dbatchsize
  * Dlayer thickness
  * Droad width
 */

 import java.io.*;

 public class clsReadFile
 {
      //global variables
      double machineNumber = 0;
      int machineType = 0;
      double DlayerThick = 0.0;
      double DroadWidth = 0.0;
      double DbatchSize = 0;

      public clsReadFile()
      {

            try
            {

                  File inputfile = new File("input.txt");
                  FileInputStream inputstream = new
                      FileInputStream(inputfile);
                  BufferedReader in = new BufferedReader(new
                      InputStreamReader(inputstream));

                  //read # of machines
                  String NM = in.readLine();
                  Double numberofmachines = new Double(NM);
                  machineNumber = numberofmachines.doubleValue();
                  //read machine type
```

```
            String MT = in.readLine();
            Integer machinename = new Integer(MT);
            machineType = machinename.intValue();
            //read batchsize
            String dBS = in.readLine();
            Double dbatchsize = new Double(dBS);
            DbatchSize = dbatchsize.doubleValue();
            //read road width
            String dRW = in.readLine();
            Double droadwidth = new Double(dRW);
            DroadWidth = droadwidth.doubleValue();
            //read layer thickness
            String dLT = in.readLine();
            Double dlayerthickness = new Double(dLT);
            DlayerThick = dlayerthickness.doubleValue();

            in.close();
      }
      catch (IOException e)
      {
            System.err.println("FileStream ERROR " + e);
      }

} //end of clsReadFile constructor

//===== FUNCTION =====//

//---- FUNCTION: getNMachines() ----//
public double getNMachines()
{
      return machineNumber;
} //end of getNMachines()

//---- FUNCTION: getMachineType() ----//
public int getMachineType()
{
      return machineType;
} //end of getMachineType()

//---- FUNCTION: getDBS() ----//
public double getDBS()
{
      return DbatchSize;
} //end of getDBS()

//---- FUNCTION: getDLT() ----//
public double getDLT()
{
      return DlayerThick;
} //end of getDLT()

//---- FUNCTION: getDRW() ----//
public double getDRW()
{
      return DroadWidth;
} //end of getDRW()
```

```
        //== END OF FUNCTIONS ==//

    } //end of clsReadFile
```

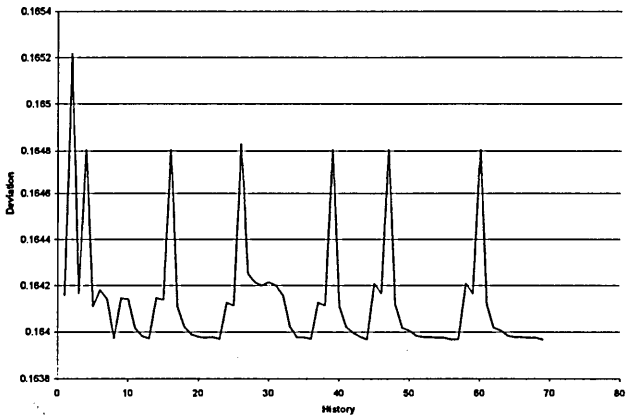
APPENDIX D

SOLUTION CONVERGENCE HISTORY OF HEARING AID EXAMPLE DATA FOR EACH MACHINE TYPE

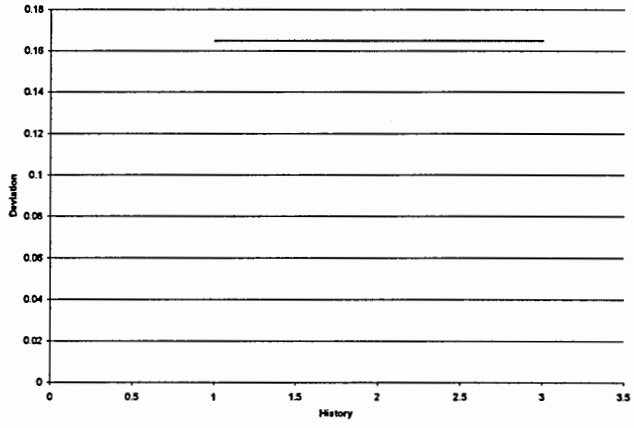
As presented in Chapter 5, Sequential Quadratic Programming is used to identify the best combination of values for the ranges of modes of managing customization. Several different starting points are used in the anaylsis to test the validity of the results. The convergence history for each starting point is presented in this appendix, organized by machine type/quantity combination.

D.1 ONE FDM MAXUM

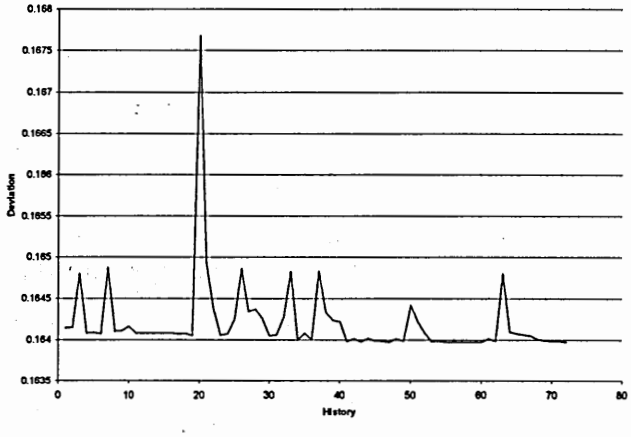
1				
		INITIAL		FINAL
1	dBS	5		5
	dLT	50		21.487
	dRW	150		21.487
	E(U)			0.8360285
2	dBS	5		5
	dLT	21.487		20.99862
	dRW	21.487		20.99862
	E(U)	0.8360285		0.8360314
3	dBS	5		5
	dLT	20.99862		20.99862
	dRW	20.99862		20.99862
	E(U)	0.8360314		0.8360314



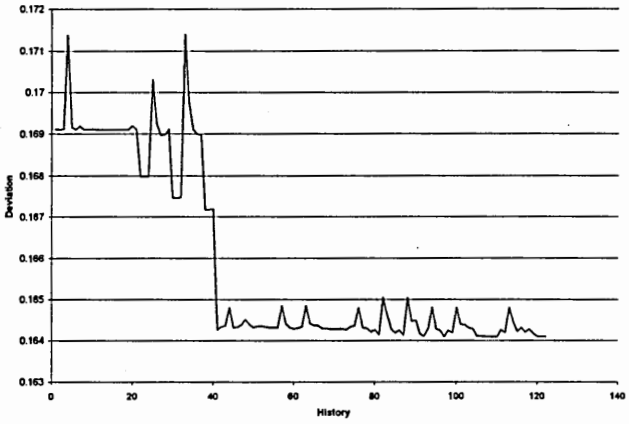
2				
		INITIAL		FINAL
1	dBS	5		5
	dLT	5		5
	dRW	5		5
	E(U)			0.8351952



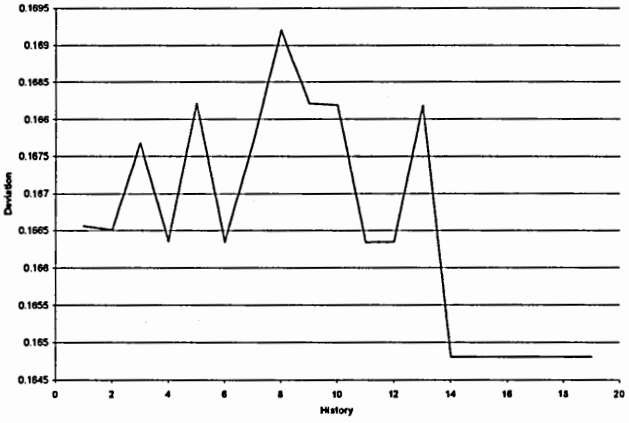
3				
		INITIAL		FINAL
1	dBS	5		5
	dLT	80		65.078
	dRW	85		69.72394
	E(U)			0.8359281
2	dBS	5		5
	dLT	65.078		34.7409
	dRW	69.72394		86.673526
	E(U)	0.8359281		0.8360261
3	dBS	5		5
	dLT	34.7409		34.7409
	dRW	86.673526		86.673526
	E(U)	0.8360261		0.8360262



4				
		INITIAL		FINAL
1	dBS	5		5
	dLT	300		302.95
	dRW	400		400
	E(U)			0.8308957
2	dBS	5		5
	dLT	302.95		19.774028
	dRW	400		87.525144
	E(U)	0.8308957		0.8357399
3	dBS	5		5
	dLT	19.774028		15.281648
	dRW	87.525144		76.41665
	E(U)	0.8357399		0.835911
4	dBS	5		5
	dLT	15.281648		15.281648
	dRW	76.41665		76.41665
	E(U)	0.835911		0.835911

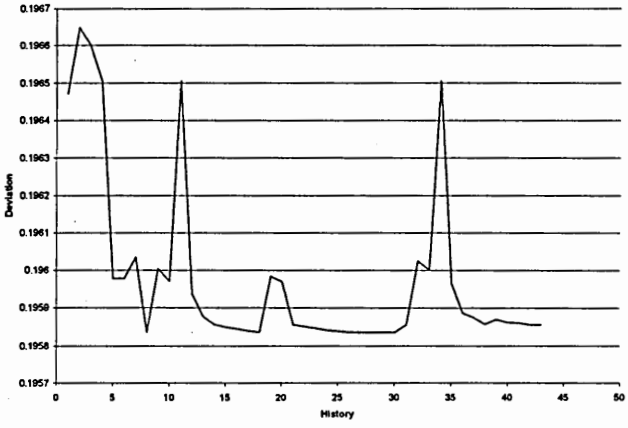


5				
		INITIAL		FINAL
1	dBS	5		5
	dLT	25		5
	dRW	350		5
	E(U)			0.8351953
2	dBS	5		5
	dLT	5		5
	dRW	5		5
	E(U)	0.8351953		0.8351953

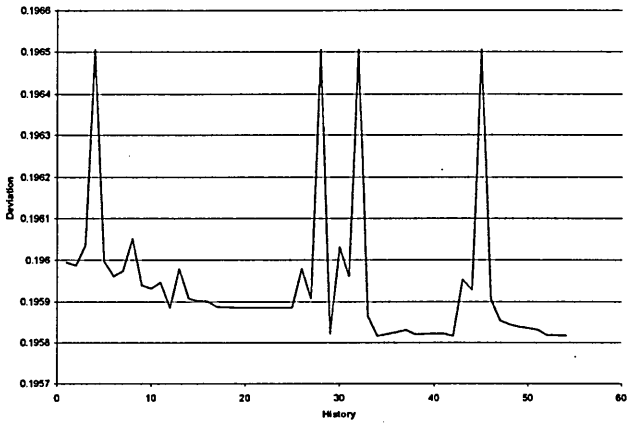


D.2 TWO FDM MAXUMS

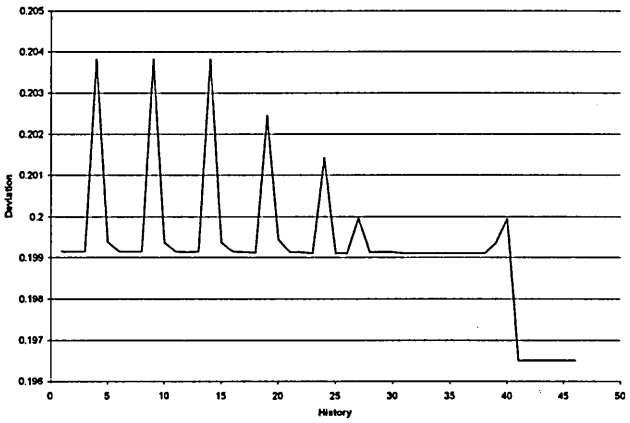
1				
		INITIAL		FINAL
1	dBS	5		5
	dLT	50		21.55
	dRW	150		21.55
	E(U)			0.8041455
2	dBS	5		5
	dLT	21.55		21.55
	dRW	21.55		21.55
	E(U)	0.8041455		0.8041455



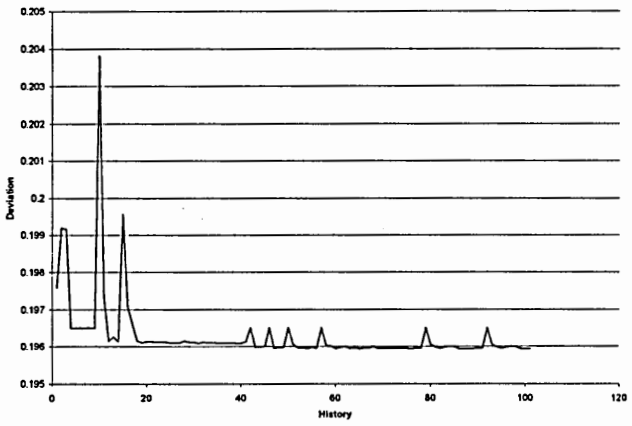
2				
		INITIAL		FINAL
1	dBS	5		5
	dLT	80		61.913418
	dRW	85		61.913418
	E(U)			0.8041138
2	dBS	5		5
	dLT	61.913418		27.69537
	dRW	61.913418		27.69537
	E(U)	0.8041138		0.8041841
3	dBS	5		5
	dLT	27.69537		27.69537
	dRW	27.69537		27.69537
	E(U)	0.8041841		0.8041841



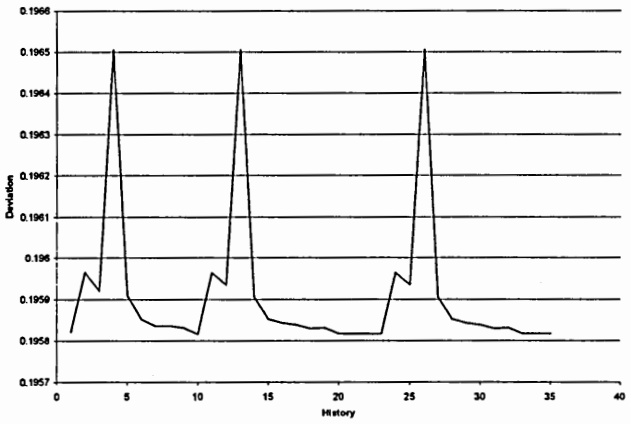
3				
		INITIAL		FINAL
1	dBS	5		5
	dLT	300		309.82348
	dRW	400		405.73605
	E(U)			0.8009061
2	dBS	5		5
	dLT	309.82348		5
	dRW	405.73605		5
	E(U)	0.8009061		0.8034949
3	dBS	5		5
	dLT	5		5
	dRW	5		5
	E(U)	0.8034949		0.8034949



4				
		INITIAL		FINAL
1	dBS	5		5
	dLT	25		5
	dRW	350		5.0435
	E(U)			0.8034949
2	dBS	5		5
	dLT	5		93.640277
	dRW	5.0435		98.227795
	E(U)	0.8034949		0.8039131
3	dBS	5		5
	dLT	93.640277		33.84198
	dRW	98.227795		35.654852
	E(U)	0.8039131		0.8040692
4	dBS	5		5
	dLT	33.84198		33.84198
	dRW	35.654852		35.654852
	E(U)	0.8040692		0.8040692

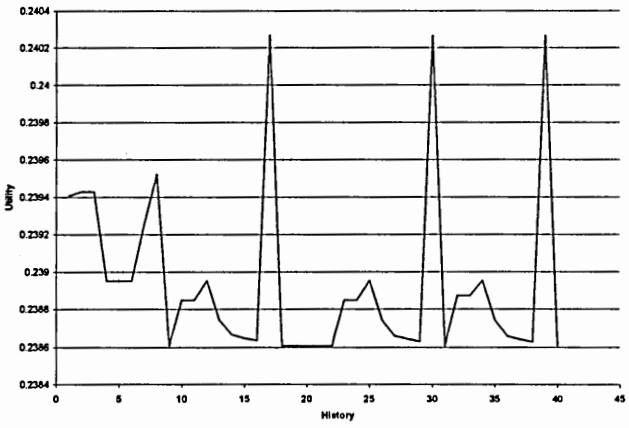


5				
		INITIAL		FINAL
1	dBS	5		5
	dLT	28		27.662914
	dRW	28		27.66291
	E(U)			0.8041832
2	dBS	5		5
	dLT	27.662914		27.662914
	dRW	27.66291		27.66291
	E(U)	0.8041832		0.8041832

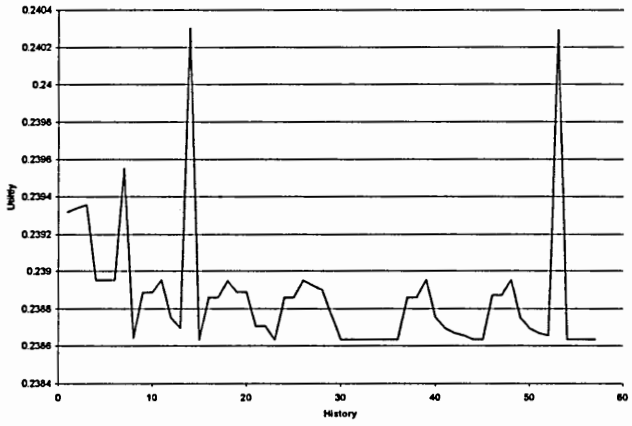


D.3 ONE FDM TITAN

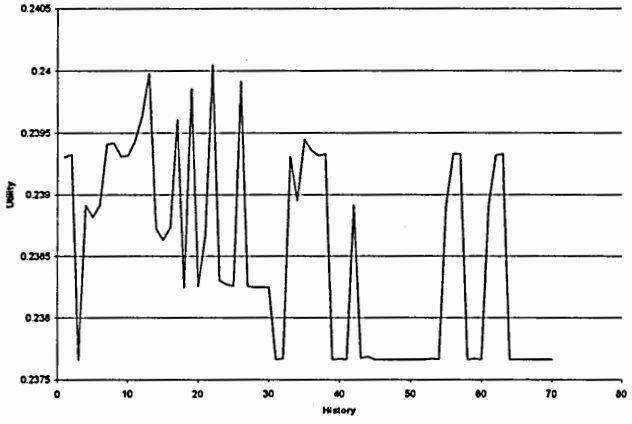
1				
		INITIAL		FINAL
1	dBS	5		5
	dLT	200		6.61869
	dRW	200		6.61869
	E(U)			0.7613919
2	dBS	5		5
	dLT	6.61869		6.61869
	dRW	6.61869		6.61869
	E(U)	0.7613919		0.7613919



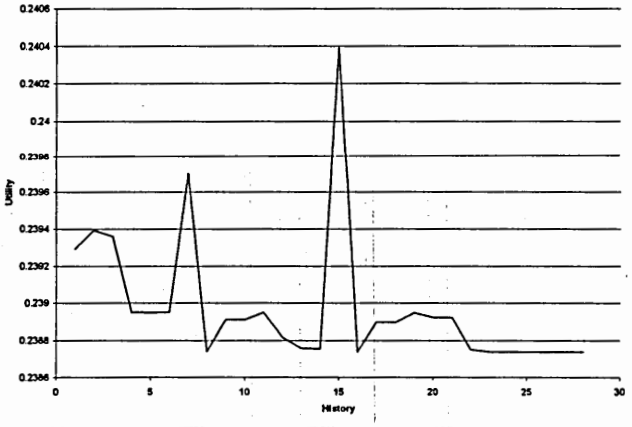
2				
		INITIAL		FINAL
1	dBS	5		5
	dLT	50		6.470463
	dRW	150		6.4705067
	E(U)			0.7613673
2	dBS	5		5
	dLT	6.470463		6.470463
	dRW	6.4705067		6.4705067
	E(U)	0.7613673		0.7613673



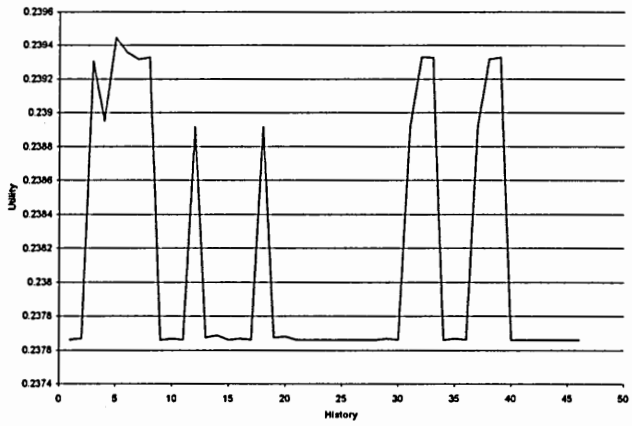
3				
		INITIAL		FINAL
1	dBS	5		5
	dLT	80		80
	dRW	85		92.999998
	E(U)			0.7623383
2	dBS	5		5
	dLT	80		79.91363
	dRW	92.999998		92.89853
	E(U)	0.7623383		0.7623395
3	dBS	5		5
	dLT	79.91363		79.91363
	dRW	92.89853		92.89853
	E(U)	0.7623395		0.7623395



4				
		INITIAL		FINAL
1	dBS	5		5
	dLT	100		5.917694
	dRW	100		5.917694
	E(U)			0.7612611

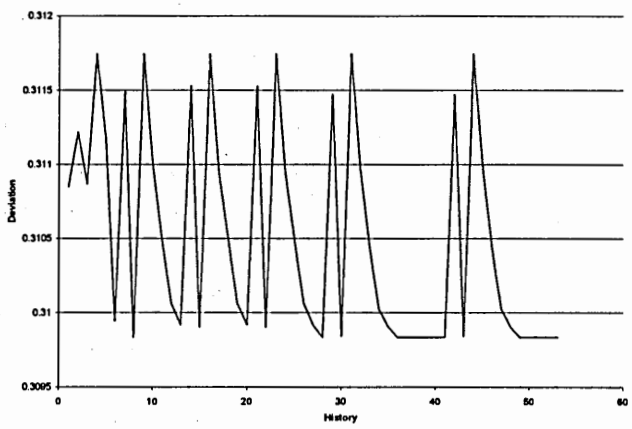


5				
		INITIAL		FINAL
1	dBS	5		5
	dLT	80		79.913139
	dRW	93		92.898
	E(U)			0.7623395
2	dBS	5		5
	dLT	79.913139		79.913139
	dRW	92.898		92.898
	E(U)	0.7623395		0.7623395

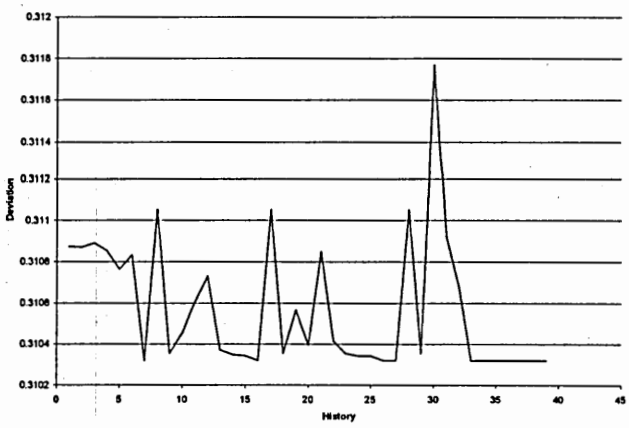


D.4 TWO FDM TITANS

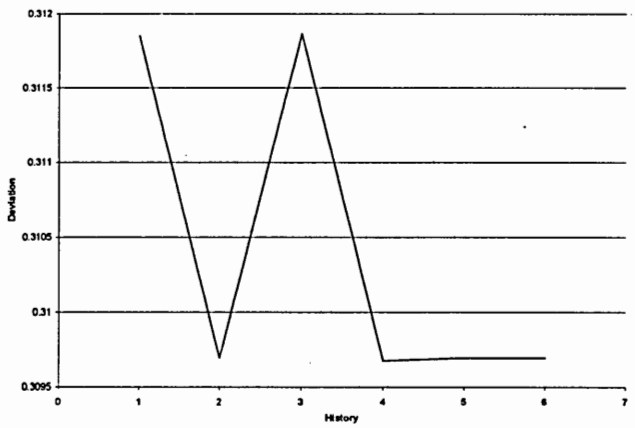
8				
		INITIAL		FINAL
1	dBS	5		5
	dLT	125		120.027
	dRW	125		180.419
	E(U)			0.6901647
2	dBS	5		5
	dLT	120.027		120.027
	dRW	180.419		180.419
	E(U)	0.6901647		0.6901647



9				
		INITIAL		FINAL
1	dBS	5		5
	dLT	50		92.492
	dRW	150		92.492
	E(U)			0.6896799
2	dBS	5		5
	dLT	92.492		92.492
	dRW	92.492		92.492
	E(U)	0.6896799		0.6896799

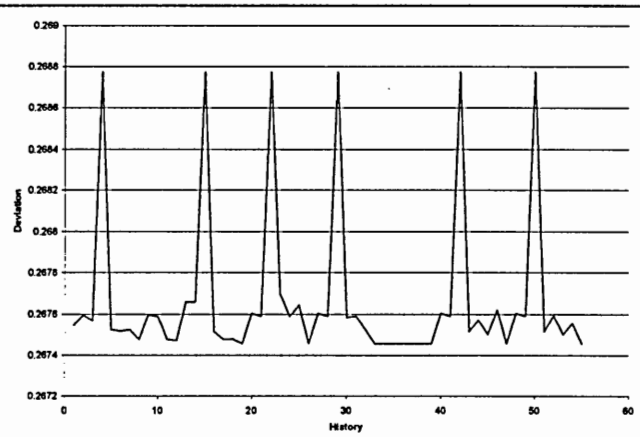


10				
		INITIAL		FINAL
1	dBS	5		5
	dLT	250		440
	dRW	250		440
	E(U)			0.6903245

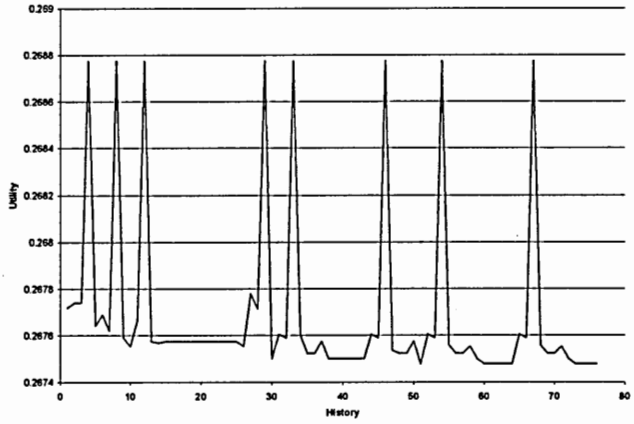


D.5 ONE PRODIGY PLUS

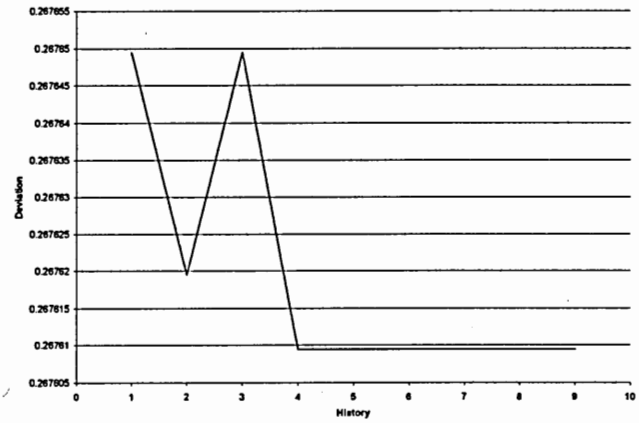
1				
		INITIAL		FINAL
1	dBS	5		5
	dLT	50		43.503974
	dRW	150		43.625474
	E(U)			0.7325426
2	dBS	5		5
	dLT	43.503974		43.503974
	dRW	43.625474		43.625474
	E(U)	0.7325426		0.7325426



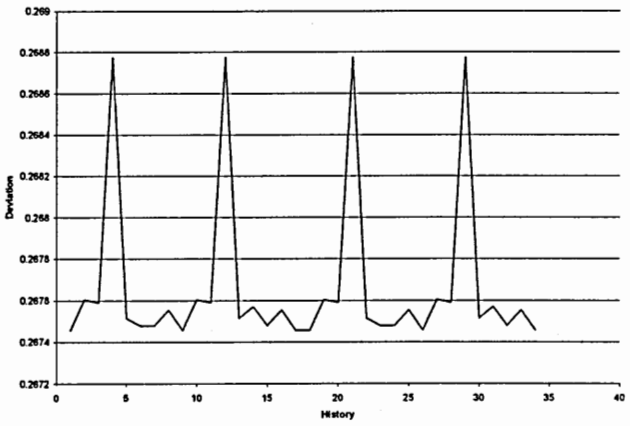
2				
		INITIAL		FINAL
1	dBS	5		5
	dLT	80		67.570157
	dRW	85		68.237854
	E(U)			0.732446
2	dBS	5		5
	dLT	67.570157		43.791297
	dRW	68.237854		44.205246
	E(U)	0.732446		0.7324986
3	dBS	5		5
	dLT	43.791297		43.55172
	dRW	44.205246		43.963118
	E(U)	0.7324986		0.7325206
4	dBS	5		5
	dLT	43.55172		43.55172
	dRW	43.963118		43.963118
	E(U)	0.7325206		0.7325206



4				
		INITIAL		FINAL
1	dBS	5		5
	dLT	25		440
	dRW	400		440
	E(U)	350		0.7323906
2	dBS	5		5
	dLT	440		440
	dRW	440		440
	E(U)	0.7323906		0.7323906

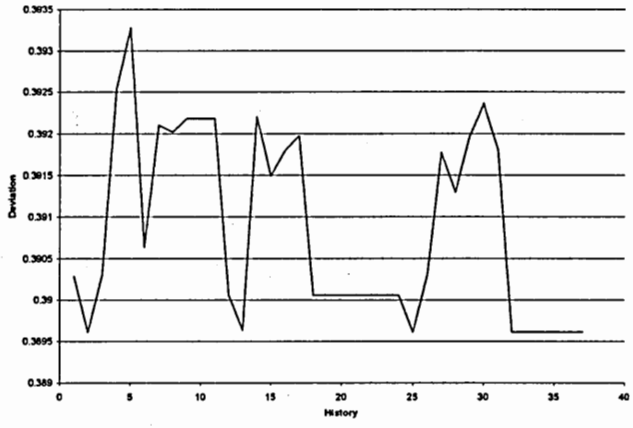


6				
		INITIAL		FINAL
1	dBS	5		5
	dLT	44		43.999
	dRW	44		43.999
	E(U)			0.7325426
2	dBS	5		5
	dLT	43.999		43.999
	dRW	43.999		43.999
	E(U)	0.7325426		0.7325426

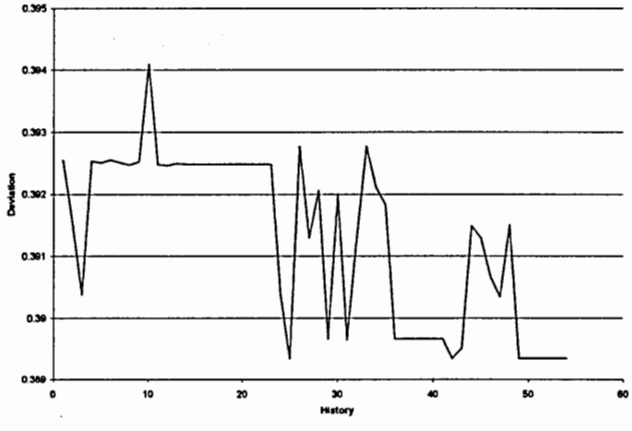


D.6 TWO PRODIGY PLUS

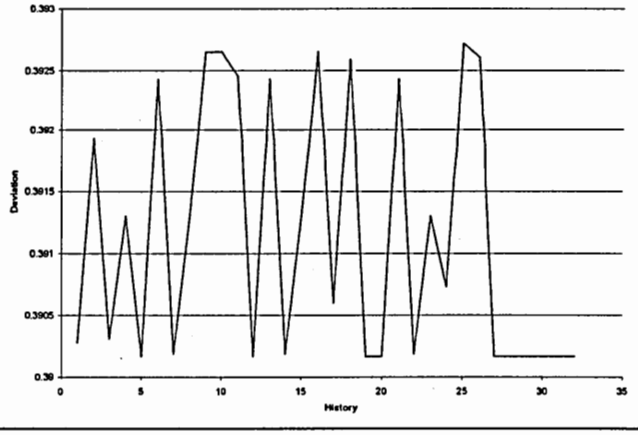
1				
		INITIAL		FINAL
1	dBS	5		5
	dLT	50		72.49982
	dRW	150		149.99855
	E(U)			0.610386
2	dBS	5		5
	dLT	72.49982		72.49982
	dRW	149.99855		149.99855
	E(U)	0.610386		0.610386



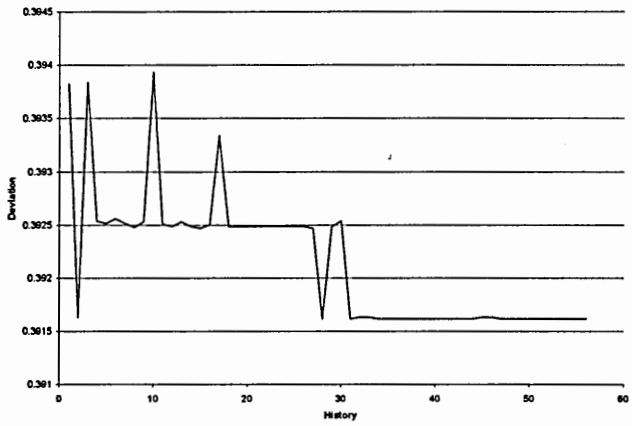
2				
		INITIAL		FINAL
1	dBS	5		5
	dLT	80		80
	dRW	85		124.9997
	E(U)			0.6096185
2	dBS	5		5
	dLT	80		117.5
	dRW	124.9997		124.997
	E(U)	0.6096185		0.6106562
3	dBS	5		5
	dLT	117.5		117.5
	dRW	124.997		124.997
	E(U)	0.6106562		0.6106562



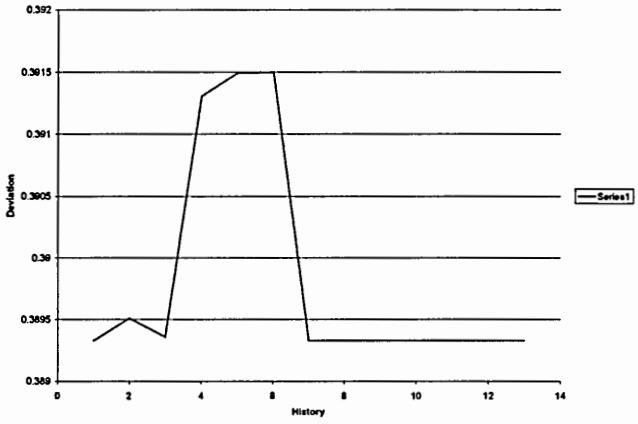
4				
		INITIAL		FINAL
1	dBS	5		5
	dLT	25		17.366382
	dRW	30		362.31434
	E(U)			0.6098335
2	dBS	5		5
	dLT	17.366382		17.366382
	dRW	362.31434		362.31434
	E(U)	0.6098335		0.6098335



5				
		INITIAL		FINAL
1	dBS	5		5
	dLT	200		218.7356
	dRW	200		440
	E(U)			0.6075294
2	dBS	5		5
	dLT	218.7356		325.6034
	dRW	440		440
	E(U)	0.6075294		0.6083855
3	dBS	5		5
	dLT	325.6034		325.6034
	dRW	440		440
	E(U)	0.6083855		0.6083855



6				
		INITIAL		FINAL
1	dBS	5		5
	dLT	125		125
	dRW	125		125
	E(U)			0.610668



APPENDIX E

MAPPING PRODUCT SPECIFICATIONS ONTO THE SPACE OF CUSTOMIZATION FOR EACH DEMAND SCENARIO OF THE PRESSURE VESSEL EXAMPLE

The results of applying the augmented PPCTM to each machine type / quantity combination are presented in Table 5.7. The specific mapping of selected ranges for the modes of managing process customization to the process parameters for one FDM Maxum is presented in Table 5.8. In this appendix, similar "roadmaps" are provided for each different machine type/quantity combination.

E.1 TWO FDM MAXUMS

Capacity	Batch Size	RW (mm)	LT (mm)	Cost ($)	Time (s)	Qual (mm)	E[U]
120	794.18	0.193	0.13192	14962.621	49914.691	0.32492	0.9129942
125	794.18			14962.621	49914.691	0.32492	0.9129942
130	827.84			15081.942	51850.118	0.32492	0.9111268
135	844.67			15202.951	53814.495	0.32492	0.9092266
140	878.33			15322.233	55749.256	0.32492	0.9073483
145	911.99			15491.532	58584.31	0.32492	0.9046446
150	945.65		0.14668	15249.89	54258.298	0.33968	0.9032913
155	979.31			15358.948	56016.776	0.33968	0.9015755
160	1012.97			15468.018	57775.459	0.33968	0.8998548
165	1029.8			15578.436	59557.254	0.33968	0.8981073
170	1063.46			15687.483	61315.524	0.33968	0.8963772
175	1080.29			15847.83	63996.109	0.33968	0.8937944
180	1130.78		0.15652	15696.464	61222.785	0.34952	0.8927464
185	1147.61			15800.965	62902.46	0.34952	0.8910852
190	1181.27			15904.252	64561.317	0.34952	0.8894393
195	1214.93			16007.548	66220.349	0.34952	0.8877889
200	1231.76			16161.97	68798.659	0.34952	0.8852824
205	1265.42		0.16636	16005.592	65935.346	0.35936	0.8843116
210	1299.08			16103.808	67506.584	0.35936	0.8827404
215	1332.74			16202.032	69077.973	0.35936	0.8811652
220	1349.57			16301.27	70666.753	0.35936	0.8795692
225	1383.23			16399.484	72237.961	0.35936	0.8779861
230	1400.06			16548.684	74726.077	0.35936	0.8755413
235	1433.72		0.1762	16383.761	71711.514	0.3692	0.8747124
240	1467.38			16477.469	73204.793	0.3692	0.8732008
245	1501.04			16571.184	74698.206	0.3692	0.8716855
250	1517.87			16665.799	76207.03	0.3692	0.8701515
255	1551.53			16809.507	78600.325	0.3692	0.867782
260	1602.02	0.22388	0.17128	16279.236	69427.835	0.39516	0.8669268
265	1635.68			16361.944	70733.966	0.39516	0.8656005
270	1652.51			16445.476	72054.236	0.39516	0.8642578
275	1686.17			16528.177	73360.253	0.39516	0.8629259
280	1703			16611.684	74680.085	0.39516	0.8615778
285	1703			16745.993	76913.658	0.39516	0.8593682
290	1703		0.1762	16671.715	75466.004	0.40008	0.8588065
295	1703			16754.151	76767.142	0.40008	0.8574714
300	1703			16836.587	78068.28	0.40008	0.8561335
305	1703			16919.024	79369.418	0.40008	0.8547928
310	1703			17051.46	81570.556	0.40008	0.8525975
315	1703		0.18604	16866.916	78209.127	0.40992	0.8520238
320	1703			16945.908	79450.542	0.40992	0.85074
325	1703			17024.9	80691.956	0.40992	0.8494536
330	1703			17103.892	81933.371	0.40992	0.8481646
335	1703			17182.885	83174.786	0.40992	0.846873
340	1703			17311.877	85316.201	0.40992	0.8447214
345	1703		0.19096	17231.336	83756.422	0.41484	0.8442647
350	1703			17308.741	84970.283	0.41484	0.8429959
355	1703			17386.146	86184.145	0.41484	0.8417245
360	1703			17463.551	87398.006	0.41484	0.8404506
365	1703			17590.955	89511.867	0.41484	0.8383108
370	1703	0.25476	0.18604	17045.616	80112.359	0.4408	0.8375395
375	1703			17115.281	81194.959	0.4408	0.8364066
380	1703			17184.946	82277.558	0.4408	0.8352716
385	1703			17254.61	83360.158	0.4408	0.8341346
390	1703			17324.275	84442.757	0.4408	0.8329955
395	1703			17443.94	86425.357	0.4408	0.8309946
400	1703		0.19096	17353.469	84692.673	0.44572	0.8306675
405	1703			17421.758	85751.331	0.44572	0.829549
410	1703			17490.046	86809.99	0.44572	0.8284286
415	1703			17558.334	87868.648	0.44572	0.8273063
420	1703			17676.622	89827.306	0.44572	0.825317
425	1703		0.19588	17583.855	88053.204	0.45064	0.8250319
430	1703			17650.836	89089.124	0.45064	0.8239292
435	1703			17717.818	90125.044	0.45064	0.8228247
440	1703			17784.8	91160.964	0.45064	0.8217183
445	1703			17851.781	92196.884	0.45064	0.8206099
450	1703			17968.763	94132.804	0.45064	0.8186284
455	1703		0.20572	17765.274	90427.262	0.46048	0.8183146
460	1703			17829.833	91420.968	0.46048	0.8172486
465	1703			17894.392	92414.674	0.46048	0.8161809
470	1703			17958.951	93408.38	0.46048	0.8151114

540	1703			18138.001	95012.176	0.49136	0.8001945
545	1703			18195.85	95891.918	0.49136	0.7992342
550	1703		0.20572	18253.698	96771.661	0.49136	0.7982724
555	1703			18311.546	97651.403	0.49136	0.7973092
560	1703			18419.394	99431.146	0.49136	0.7954639
565	1703			18315.407	97455.24	0.49628	0.7953397
570	1703			18372.265	98317.676	0.49628	0.7943923
575	1703		0.21064	18429.124	99180.112	0.49628	0.7934435
580	1703			18485.982	100042.55	0.49628	0.7924934
585	1703			18542.84	100904.98	0.49628	0.7915418
590	1703			18649.699	102667.42	0.49628	0.7897032
595	1703			18544.268	100664.38	0.5012	0.7896082
600	1703			18600.183	101510.3	0.5012	0.7886719
605	1703		0.21556	18656.098	102356.22	0.5012	0.7877343
610	1703			18712.012	103202.14	0.5012	0.7867953
615	1703			18817.927	104948.06	0.5012	0.7849656
620	1703			18712.167	102937.32	0.50612	0.7848776
625	1703			18767.181	103767.46	0.50612	0.7839532
630	1703		0.22048	18822.195	104597.6	0.50612	0.7830275
635	1703	0.28564		18877.209	105427.74	0.50612	0.7821005
640	1703			18932.223	106257.88	0.50612	0.7811722
645	1703			19037.238	107988.02	0.50612	0.7793485
650	1703			18930.397	105956.38	0.51104	0.7792833
655	1703			18984.551	106771.43	0.51104	0.778369
660	1703		0.2254	19038.704	107586.48	0.51104	0.7774535
665	1703			19092.858	108401.53	0.51104	0.7765368
670	1703			19197.012	110116.58	0.51104	0.7747211
675	1703			19090.086	108081.44	0.51596	0.7746583
680	1703			19143.417	108882.04	0.51596	0.7737551
685	1703		0.23032	19196.748	109682.64	0.51596	0.7728506
690	1703			19250.079	110483.25	0.51596	0.7719449
695	1703			19303.41	111283.85	0.51596	0.771038
700	1703			19406.74	112984.45	0.51596	0.7692275
705	1703			19299.025	110933.39	0.52088	0.7691823
710	1703			19351.568	111720.15	0.52088	0.7682884
715	1703		0.23524	19404.111	112506.91	0.52088	0.7673933
720	1703			19456.655	113293.67	0.52088	0.7664969
725	1703			19559.198	114980.43	0.52088	0.7646938
730	1703			19065.933	106624.71	0.54192	0.7646613
735	1703			19115.081	107355.02	0.54192	0.7638322
740	1703			19164.228	108085.32	0.54192	0.7630022
745	1703			19213.376	108815.63	0.54192	0.7621711
750	1703			19262.523	109545.94	0.54192	0.7613389
755	1703		0.2254	19361.67	111176.24	0.54192	0.7596048
760	1703			19360.818	111006.55	0.54192	0.7596715
765	1703			19409.965	111736.86	0.54192	0.7588362
770	1703			19459.113	112467.16	0.54192	0.7579998
775	1703			19508.26	113197.47	0.54192	0.7571624
780	1703			19607.408	114827.77	0.54192	0.7554175
785	1703			19491.656	112634.92	0.54684	0.7554865
790	1703			19540.072	113352.34	0.54684	0.7546617
795	1703		0.23032	19588.487	114069.76	0.54684	0.7538358
800	1703			19636.903	114787.18	0.54684	0.753009
805	1703			19685.318	115504.6	0.54684	0.752181
810	1703			19783.734	117122.02	0.54684	0.7504418
815	1703			19667.985	114926.83	0.55176	0.7505132
820	1703			19715.7	115631.9	0.55176	0.7496973
825	1703	0.31652	0.23524	19763.415	116336.98	0.55176	0.7488805
830	1703			19811.13	117042.05	0.55176	0.7480627
835	1703			19858.845	117747.12	0.55176	0.7472439
840	1703			19956.561	119352.19	0.55176	0.7455099
845	1703			19840.896	117156.07	0.55668	0.745582
850	1703			19887.94	117849.3	0.55668	0.7447749
855	1703		0.24016	19934.985	118542.53	0.55668	0.7439667
860	1703			19982.029	119235.76	0.55668	0.7431576
865	1703			20079.073	120828.99	0.55668	0.7414307
870	1703			19964.209	118644.55	0.5616	0.7414878
875	1703			20010.61	119326.41	0.5616	0.7406899
880	1703			20057.011	120008.28	0.5616	0.739891
885	1703			20103.412	120690.14	0.5616	0.7390912
890	1703			20149.814	121372.01	0.5616	0.7382905
895	1703		0.24508	20246.215	122953.87	0.5616	0.7365682

E.2 ONE FDM TITAN

Capacity	Batch Size	RW (mm)	LT (mm)	Cost ($)	Time (s)	Qual (mm)	E[U]	Constraint
120	714.26			8291.7833	107483	0.45	0.9124272	601179.51
125	714.26			8291.7833	107483	0.45	0.9124272	601179.51
130	738.2			8421.2866	111689.2	0.45	0.9098929	602308.28
135	762.14			8550.7581	115894.31	0.45	0.9073478	603663.91
140	778.1			8681.0413	120127.25	0.45	0.9047747	600326.11
145	802.04			8810.4368	124329.76	0.45	0.9022081	602234.97
150	810.02			8941.4558	128587.92	0.45	0.8995965	595159.44
155	810.02		0.24	9073.2943	132874.19	0.45	0.896956	584395.01
160	810.02			9205.1327	137160.45	0.45	0.8943034	574571.25
165	810.02			9336.9712	141446.72	0.45	0.8916385	565602.64
170	810.02			9468.8097	145732.98	0.45	0.8889614	557413.73
175	810.02			9600.6482	150019.24	0.45	0.886272	549937.69
180	810.02			9732.4867	154305.51	0.45	0.8835702	543115.11
185	810.02			9864.3251	158591.77	0.45	0.8808559	536893
190	810.02			9996.1636	162878.04	0.45	0.8781292	531223.97
195	810.02			10178.002	168964.3	0.45	0.8742855	527865.47
200	810.02			9955.8071	160920.26	0.4692	0.8716269	510848.9
205	810.02			10080.045	164943.26	0.4692	0.8690563	506337.06
210	810.02			10254.282	170766.27	0.4692	0.8653678	504031.64
215	810.02			10328.52	172989.28	0.4692	0.8638818	498504.29
220	810.02			10452.758	177012.28	0.4692	0.8612777	495129.23
225	810.02			10576.995	181035.29	0.4692	0.8586624	492082.97
230	810.02			10701.233	185058.29	0.4692	0.8560357	489344.07
235	810.02			10825.471	189081.3	0.4692	0.8533977	486892.91
240	810.02			10949.708	193104.31	0.4692	0.8507482	484711.51
245	810.02			11073.946	197127.31	0.4692	0.8480873	482783.35
250	810.02			11198.184	201150.32	0.4692	0.8454148	481093.23
255	810.02			11322.421	205173.33	0.4692	0.8427308	479627.16
260	810.02			11446.659	209196.33	0.4692	0.8400351	478372.21
265	810.02			11570.897	213219.34	0.4692	0.8373278	477316.43
270	810.02			11695.134	217242.35	0.4692	0.8346087	476448.75
275	810.02			11869.372	223065.35	0.4692	0.8307066	477558.91
280	810.02			11943.609	225288.36	0.4692	0.8291352	475237.39
285	810.02			12067.847	229311.37	0.4692	0.8263807	474875.32
290	810.02			12192.085	233334.37	0.4692	0.8236141	474664.47
295	810.02			12316.322	237357.38	0.4692	0.8208356	474597.13
300	810.02			12440.56	241380.38	0.4692	0.8180451	474666.14
305	810.02			12614.798	247203.39	0.4692	0.8140401	476664.79
310	810.02			12689.035	249426.4	0.4692	0.8124276	475186.81
315	810.02			12813.273	253449.4	0.4692	0.8096006	475626.32
320	810.02			12937.511	257472.41	0.4692	0.8067614	476177.81
325	810.02			13061.748	261495.42	0.4692	0.8039098	476836.12
330	810.02			13185.986	265518.42	0.4692	0.8010458	477596.39
335	810.02			13310.224	269541.43	0.4692	0.7981694	478454.05
340	810.02			13434.461	273564.44	0.4692	0.7952806	479404.81
345	810.02			13558.699	277587.44	0.4692	0.7923792	480444.63
350	810.02			13682.936	281610.45	0.4692	0.7894652	481569.67
355	810.02			13857.174	287433.46	0.4692	0.7852825	484576.35
360	810.02			13931.412	289656.46	0.4692	0.7835992	484061.26
365	810.02			14055.649	293679.47	0.4692	0.780647	485421.19
370	810.02			14179.887	297702.47	0.4692	0.7776821	486853.09
375	810.02			14304.125	301725.48	0.4692	0.7747043	488354.09
380	810.02			14428.362	305748.49	0.4692	0.7717135	489921.46
385	810.02			14552.6	309771.49	0.4692	0.7687098	491552.61
390	810.02			14676.838	313794.5	0.4692	0.765693	493245.08
395	810.02	0.21		14851.075	319617.51	0.4692	0.7613624	496796.56
400	810.02			14925.313	321840.51	0.4692	0.7596201	496804.83
405	810.02			15049.551	325863.52	0.4692	0.7565638	498667.79
410	810.02			15173.788	329886.53	0.4692	0.7534942	500583.42
415	810.02			15298.026	333909.53	0.4692	0.7504113	502549.84
420	810.02			15422.263	337932.54	0.4692	0.747315	504565.22
425	810.02			15546.501	341955.55	0.4692	0.7442052	506627.85
430	810.02			15670.739	345978.55	0.4692	0.741082	508736.06
435	810.02		0.2592	15844.976	351801.56	0.4692	0.7365981	512688.29
440	810.02			15919.214	354024.56	0.4692	0.7347946	513083.04
445	810.02			16043.452	358047.57	0.4692	0.7316303	515318.87
450	810.02			16167.689	362070.58	0.4692	0.7284524	517594.42
455	810.02			16291.927	366093.58	0.4692	0.7252606	519908.37
460	810.02			16416.165	370116.59	0.4692	0.7220549	522259.48
465	810.02			16540.402	374139.6	0.4692	0.7188352	524646.54
470	810.02			16664.64	378162.6	0.4692	0.7156016	527068.41

E.3 TWO FDM TITANS

Capacity	Batch Size	RW (mm)	LT (mm)	Cost ($)	Time (s)	Qual (mm)	E[U]
120	802.04			13079.24	50299.511	0.4692	0.8780461
125	802.04			13079.24	50299.511	0.4692	0.8780461
130	810.02			13202.781	52299.083	0.4692	0.8761824
135	810.02			13327.019	54310.587	0.4692	0.8743021
140	810.02			13451.257	56322.09	0.4692	0.8724157
145	810.02			13575.494	58333.593	0.4692	0.8705234
150	810.02			13699.732	60345.096	0.4692	0.8686249
155	810.02			13823.97	62356.599	0.4692	0.8667205
160	810.02			13948.207	64368.103	0.4692	0.8648099
165	810.02			14072.445	66379.606	0.4692	0.8628932
170	810.02			14196.683	68391.109	0.4692	0.8609704
175	810.02			14320.92	70402.612	0.4692	0.8590414
180	810.02			14445.158	72414.115	0.4692	0.8571063
185	810.02			14569.396	74425.619	0.4692	0.855165
190	810.02			14693.633	76437.122	0.4692	0.8532175
195	810.02			14817.871	78448.625	0.4692	0.8512637
200	810.02			14942.109	80460.128	0.4692	0.8493037
205	810.02			15066.346	82471.631	0.4692	0.8473375
210	810.02			15190.584	84483.135	0.4692	0.8453649
215	810.02			15314.821	86494.638	0.4692	0.8433861
220	810.02			15439.059	88506.141	0.4692	0.8414009
225	810.02			15563.297	90517.644	0.4692	0.8394093
230	810.02			15687.534	92529.147	0.4692	0.8374114
235	810.02			15811.772	94540.651	0.4692	0.8354071
240	810.02			15936.01	96552.154	0.4692	0.8333964
245	810.02			16060.247	98563.657	0.4692	0.8313793
250	810.02			16184.485	100575.16	0.4692	0.8293556
255	810.02			16308.723	102586.66	0.4692	0.8273256
260	810.02			16432.96	104598.17	0.4692	0.825289
265	810.02			16557.198	106609.67	0.4692	0.8232459
270	810.02			16681.436	108621.17	0.4692	0.8211962
275	810.02			16805.673	110632.68	0.4692	0.81914
280	810.02			16929.911	112644.18	0.4692	0.8170772
285	810.02			17054.148	114655.68	0.4692	0.8150078
290	810.02			17178.386	116667.19	0.4692	0.8129318
295	810.02			17302.624	118678.69	0.4692	0.8108491
300	810.02			17426.861	120690.19	0.4692	0.8087598
305	810.02			17551.099	122701.7	0.4692	0.8066637
310	810.02			17675.337	124713.2	0.4692	0.804561
315	810.02			17799.574	126724.7	0.4692	0.8024515
320	810.02			17923.812	128736.21	0.4692	0.8003352
325	810.02			18048.05	130747.71	0.4692	0.7982122
330	810.02	0.21	0.2592	18172.287	132759.21	0.4692	0.7960824
335	810.02			18296.525	134770.71	0.4692	0.7939457
340	810.02			18420.763	136782.22	0.4692	0.7918022
345	810.02			18545	138793.72	0.4692	0.7896518
350	810.02			18669.238	140805.22	0.4692	0.7874946
355	810.02			18793.475	142816.73	0.4692	0.7853304
360	810.02			18917.713	144828.23	0.4692	0.7831592
365	810.02			19041.951	146839.73	0.4692	0.7809812
370	810.02			19166.188	148851.24	0.4692	0.7787961
375	810.02			19290.426	150862.74	0.4692	0.776604
380	810.02			19414.664	152874.24	0.4692	0.7744049
385	810.02			19538.901	154885.75	0.4692	0.7721987
390	810.02			19663.139	156897.25	0.4692	0.7699855
395	810.02			19787.377	158908.75	0.4692	0.7677652
400	810.02			19911.614	160920.26	0.4692	0.7655377
405	810.02			20035.852	162931.76	0.4692	0.7633031
410	810.02			20160.09	164943.26	0.4692	0.7610613
415	810.02			20284.327	166954.77	0.4692	0.7588123
420	810.02			20408.565	168966.27	0.4692	0.7565561
425	810.02			20532.802	170977.77	0.4692	0.7542927
430	810.02			20657.04	172989.28	0.4692	0.752022
435	810.02			20781.278	175000.78	0.4692	0.749744
440	810.02			20905.515	177012.28	0.4692	0.7474587
445	810.02			21029.753	179023.79	0.4692	0.7451661
450	810.02			21153.991	181035.29	0.4692	0.7428661
455	810.02			21278.228	183046.79	0.4692	0.7405587
460	810.02			21402.466	185058.29	0.4692	0.7382439
465	810.02			21526.704	187069.8	0.4692	0.7359217
470	810.02			21650.941	189081.3	0.4692	0.733592

540	810.02			23390.268	217242.35	0.4692	0.7001795
545	810.02			23514.506	219253.85	0.4692	0.697735
550	810.02			23638.744	221265.35	0.4692	0.6952825
555	810.02			23762.981	223276.86	0.4692	0.6928223
560	810.02			23937.219	226188.36	0.4692	0.6893239
565	810.02			24011.457	227299.86	0.4692	0.6878779
570	810.02			24135.694	229311.37	0.4692	0.6853938
575	810.02			24259.932	231322.87	0.4692	0.6829017
580	810.02			24384.169	233334.37	0.4692	0.6804016
585	810.02			24508.407	235345.88	0.4692	0.6778934
590	810.02			24632.645	237357.38	0.4692	0.6753772
595	810.02			24756.882	239368.88	0.4692	0.6728529
600	810.02			24881.12	241380.38	0.4692	0.6703204
605	810.02			25005.358	243391.89	0.4692	0.6677798
610	810.02			25129.595	245403.39	0.4692	0.665231
615	738.2			25287.315	247988.88	0.4692	0.6644976
620	810.02			25378.071	249426.4	0.4692	0.6639324
625	810.02			25502.308	251437.9	0.4692	0.6631397
630	810.02			25626.546	253449.4	0.4692	0.6623451
635	810.02			25750.784	255460.91	0.4692	0.6615485
640	810.02			25875.021	257472.41	0.4692	0.66075
645	810.02			25999.259	259483.91	0.4692	0.6599495
650	810.02			26123.496	261495.42	0.4692	0.659147
655	810.02			26247.734	263506.92	0.4692	0.6583425
660	810.02			26371.972	265518.42	0.4692	0.657536
665	810.02			26496.209	267529.93	0.4692	0.6567276
670	810.02			26620.447	269541.43	0.4692	0.6559171
675	810.02			26744.685	271552.93	0.4692	0.6551046
680	810.02			26868.922	273564.44	0.4692	0.6542902
685	810.02			26993.16	275575.94	0.4692	0.6534737
690	810.02			27117.398	277587.44	0.4692	0.6526552
695	810.02			27241.635	279598.95	0.4692	0.6518346
700	810.02			27365.873	281610.45	0.4692	0.6510121
705	810.02			27490.111	283621.95	0.4692	0.6501875
710	810.02			27614.348	285633.46	0.4692	0.6493608
715	810.02			27738.586	287644.96	0.4692	0.6485322
720	810.02			27862.824	289656.46	0.4692	0.6477014
725	810.02			27987.061	291667.96	0.4692	0.6468687
730	810.02			28111.299	293679.47	0.4692	0.6460338
735	810.02			28235.536	295690.97	0.4692	0.6451969
740	810.02			28359.774	297702.47	0.4692	0.644358
745	810.02			28484.012	299713.98	0.4692	0.6435169
750	810.02			28608.249	301725.48	0.4692	0.6426738
755	810.02			28732.487	303736.98	0.4692	0.6418286
760	810.02			28856.725	305748.49	0.4692	0.6409813
765	810.02			28980.962	307759.99	0.4692	0.6401319
770	810.02	0.21	0.2592	29105.2	309771.49	0.4692	0.6392804
775	810.02			29229.438	311783	0.4692	0.6384268
780	810.02			29353.675	313794.5	0.4692	0.6375711
785	810.02			29477.913	315806	0.4692	0.6367133
790	810.02			29602.151	317817.51	0.4692	0.6358533
795	810.02			29726.388	319829.01	0.4692	0.6349912
800	810.02			29850.626	321840.51	0.4692	0.634127
805	810.02			29974.863	323852.02	0.4692	0.6332607
810	810.02			30099.101	325863.52	0.4692	0.6323922
815	810.02			30223.339	327875.02	0.4692	0.6315216
820	810.02			30347.576	329886.53	0.4692	0.6306488
825	810.02			30471.814	331898.03	0.4692	0.6297739
830	810.02			30596.052	333909.53	0.4692	0.6288968
835	810.02			30720.289	335921.04	0.4692	0.6280175
840	810.02			30844.527	337932.54	0.4692	0.627136
845	810.02			30968.765	339944.04	0.4692	0.6262524
850	810.02			31093.002	341955.55	0.4692	0.6253666
855	810.02			31217.24	343967.05	0.4692	0.6244786
860	810.02			31341.478	345978.55	0.4692	0.6235884
865	810.02			31465.715	347990.05	0.4692	0.622696
870	810.02			31589.953	350001.56	0.4692	0.6218014
875	810.02			31714.19	352013.06	0.4692	0.6209046
880	810.02			31838.428	354024.56	0.4692	0.6200056
885	810.02			31962.666	356036.07	0.4692	0.6191043
890	810.02			32086.903	358047.57	0.4692	0.6182008
895	810.02			32211.141	360059.07	0.4692	0.6172951

E.4 ONE PRODIGY PLUS

Capacity	Batch Size	RW (mm)	LT (mm)	Cost ($)	Time (s)	Qual (mm)	E[U]
120	234	0.21	0.29352	7621.0359	176596.86	0.50352	0.8684775
125	234			7842.6886	183955.06	0.50352	0.8684775
130	234			8064.3414	191313.26	0.50352	0.8638643
135	234			8285.9941	198671.47	0.50352	0.8592145
140	234			8507.6469	206029.67	0.50352	0.8545277
145	234			8729.2996	213387.87	0.50352	0.8498038
150	234			8950.9524	220746.07	0.50352	0.8450424
155	234			9172.6052	228104.28	0.50352	0.8402431
160	234			9444.2579	237262.48	0.50352	0.834255
165	234		0.33	9004.6956	221698.1	0.54	0.8291479
170	234			9207.8267	228416.23	0.54	0.824755
175	234			9410.9578	235134.35	0.54	0.8203302
180	234			9614.0889	241852.48	0.54	0.8158734
185	234			9817.2199	248570.6	0.54	0.8113841
190	234			10020.351	255288.73	0.54	0.8068623
195	234			10223.482	262006.85	0.54	0.8023077
200	234			10426.613	268724.98	0.54	0.7977199
205	234			10679.744	277243.1	0.54	0.7918967
210	234			10832.875	282161.22	0.54	0.7884443
215	234			11036.006	288879.35	0.54	0.7837558
220	234			11239.138	295597.47	0.54	0.7790333
225	234			11442.269	302315.6	0.54	0.7742764
230	234			11645.4	309033.72	0.54	0.7694849
235	234			11848.531	315751.85	0.54	0.7646586
240	234			12051.662	322469.97	0.54	0.7597972
245	234			12254.793	329188.09	0.54	0.7549004
250	234			12507.924	337706.22	0.54	0.7486842
255	234			12661.055	342624.34	0.54	0.7449997
260	234			12864.186	349342.47	0.54	0.7399952
265	234			13067.317	356060.59	0.54	0.7349542
270	234			13270.448	362778.72	0.54	0.7298766
275	234			13473.579	369496.84	0.54	0.7247619
280	234			13676.711	376214.97	0.54	0.71961
285	234			13879.842	382933.09	0.54	0.7144206
290	234			14082.973	389651.21	0.54	0.7091933
295	234			14336.104	398169.34	0.54	0.7025568
300	234			14489.235	403087.46	0.54	0.6986241
305	234			14692.366	409805.59	0.54	0.6932816
310	234			14895.497	416523.71	0.54	0.6879002
315	234			15098.628	423241.84	0.54	0.6824795
320	234			15301.759	429959.96	0.54	0.6770192
325	234			15504.89	436678.08	0.54	0.6715191
330	234			15708.021	443396.21	0.54	0.6659788
335	234			15911.152	450114.33	0.54	0.6603981
340	234			16164.284	458632.46	0.54	0.6533122
345	234			16317.415	463550.58	0.54	0.6491141
350	234			16520.546	470268.71	0.54	0.6434101
355	234			16723.677	476986.83	0.54	0.6376645
360	234			16926.808	483704.96	0.54	0.6318769
365	234			17129.939	490423.08	0.54	0.626047
370	234			17333.07	497141.2	0.54	0.6201744
375	234			17536.201	503859.33	0.54	0.6142589
380	234			17789.332	512377.45	0.54	0.6067472
385	234			17942.463	517295.58	0.54	0.6022978
390	234			18145.594	524013.7	0.54	0.5962515
395	234			18348.725	530731.83	0.54	0.5901609
400	234			18551.857	537449.95	0.54	0.5840258
405	234			18754.988	544168.07	0.54	0.5778457
410	234			18958.119	550886.2	0.54	0.5716204
415	214.74			19230.423	559975.99	0.54	0.5631552

E.5 TWO PRODIGY PLUS

Capacity	Batch Size	RW (mm)	LT (mm)	Cost ($)	Time (s)	Qual (mm)	E[U]
120	234			9687.0387	84233.269	0.52176	0.8755902
125	234			9898.8845	87742.988	0.52176	0.8755902
130	234			10110.73	91252.708	0.52176	0.8725005
135	234			10322.576	94762.427	0.52176	0.869394
140	234			10534.422	98272.147	0.52176	0.8662707
145	234			10746.268	101781.87	0.52176	0.8631305
150	234			10958.113	105291.59	0.52176	0.8599732
155	234			11169.959	108801.31	0.52176	0.8567988
160	234			11381.805	112311.02	0.52176	0.8536071
165	234			11593.651	115820.74	0.52176	0.8503982
170	234			11805.497	119330.46	0.52176	0.8471718
175	234			12017.342	122840.18	0.52176	0.8439279
180	234		0.31176	12229.188	126349.9	0.52176	0.8406664
185	234			12441.034	129859.62	0.52176	0.8373872
190	234			12652.88	133369.34	0.52176	0.8340901
195	234			12864.726	136879.06	0.52176	0.8307752
200	234			13076.571	140388.78	0.52176	0.8274423
205	234			13288.417	143898.5	0.52176	0.8240912
210	234			13500.263	147408.22	0.52176	0.820722
215	234			13712.109	150917.94	0.52176	0.8173344
220	234			13923.954	154427.66	0.52176	0.8139284
225	234			14135.8	157937.38	0.52176	0.8105039
230	234			14347.646	161447.1	0.52176	0.8070608
235	234			14559.492	164956.82	0.52176	0.803599
240	234			14771.338	168466.54	0.52176	0.8001184
245	234			15033.183	172876.26	0.52176	0.7957644
250	234			14759.294	167953.11	0.54	0.7928524
255	234			14962.425	171312.17	0.54	0.7895009
260	234			15165.556	174671.23	0.54	0.7861319
265	234			15368.687	178030.3	0.54	0.7827454
270	234			15571.818	181389.36	0.54	0.7793412
275	234			15774.949	184748.42	0.54	0.7759193
280	234			15978.08	188107.48	0.54	0.7724795
285	234			16181.211	191466.54	0.54	0.7690218
290	234			16384.343	194825.61	0.54	0.765546
295	234			16587.474	198184.67	0.54	0.7620521
300	234			16790.605	201543.73	0.54	0.7585399
305	234			16993.736	204902.79	0.54	0.7550095
310	234	0.21		17196.867	208261.86	0.54	0.7514606
315	234			17399.998	211620.92	0.54	0.7478932
320	234			17603.129	214979.98	0.54	0.7443071
325	234			17806.26	218339.04	0.54	0.7407024
330	234			18009.391	221698.1	0.54	0.7370788
335	234			18212.522	225057.17	0.54	0.7334363
340	234			18415.653	228416.23	0.54	0.7297747
345	234			18618.784	231775.29	0.54	0.7260941
350	234			18821.916	235134.35	0.54	0.7223942
355	234			19025.047	238493.42	0.54	0.718675
360	234			19228.178	241852.48	0.54	0.7149363
365	234			19431.309	245211.54	0.54	0.7111781
370	234			19684.44	249470.6	0.54	0.7064385
375	234		0.33	19837.571	251929.66	0.54	0.7036027
380	234			20040.702	255288.73	0.54	0.6997852
385	234			20243.833	258647.79	0.54	0.6959478
390	234			20446.964	262006.85	0.54	0.6920903
395	234			20650.095	265365.91	0.54	0.6882127
400	234			20853.226	268724.98	0.54	0.6843148
405	234			21056.357	272084.04	0.54	0.6803964
410	234			21259.489	275443.1	0.54	0.6764576
415	234			21462.62	278802.16	0.54	0.6724982
420	234			21665.751	282161.22	0.54	0.668518
425	234			21868.882	285520.29	0.54	0.664517
430	234			22072.013	288879.35	0.54	0.6604951
435	234			22275.144	292238.41	0.54	0.6564521
440	234			22478.275	295597.47	0.54	0.6523879
445	234			22681.406	298956.53	0.54	0.6483025
450	234			22884.537	302315.6	0.54	0.6441957
455	234			23087.668	305674.66	0.54	0.6400673
460	234			23290.799	309033.72	0.54	0.6359173
465	234			23493.93	312392.78	0.54	0.6317456
470	234			23697.062	315751.85	0.54	0.627552

505	234			25118.979	339265.28	0.54	0.5975751
510	234			25322.11	342624.34	0.54	0.5962258
515	234			25525.241	345983.41	0.54	0.5947405
520	234			25728.372	349342.47	0.54	0.593249
525	234			25931.503	352701.53	0.54	0.5917513
530	234			26134.635	356060.59	0.54	0.5902475
535	234			26337.766	359419.65	0.54	0.5887375
540	234			26540.897	362778.72	0.54	0.5872212
545	234			26744.028	366137.78	0.54	0.5856986
550	234			26947.159	369496.84	0.54	0.5841698
555	234			27150.29	372855.9	0.54	0.5826347
560	234			27353.421	376214.97	0.54	0.5810932
565	234			27556.552	379574.03	0.54	0.5795453
570	234			27759.683	382933.09	0.54	0.577991
575	234			27962.814	386292.15	0.54	0.5764303
580	234			28165.945	389651.21	0.54	0.5748632
585	234			28369.076	393010.28	0.54	0.5732896
590	234			28572.208	396369.34	0.54	0.5717095
595	234			28775.339	399728.4	0.54	0.5701228
600	234			28978.47	403087.46	0.54	0.5685296
605	234			29181.601	406446.52	0.54	0.5669298
610	234			29384.732	409805.59	0.54	0.5653235
615	234			29587.863	413164.65	0.54	0.5637104
620	234			29840.994	417423.71	0.54	0.5616556
625	234			29994.125	419882.77	0.54	0.5604643
630	234			30197.256	423241.84	0.54	0.5588312
635	234			30400.387	426600.9	0.54	0.5571914
640	234			30603.518	429959.96	0.54	0.5555447
645	234			30806.649	433319.02	0.54	0.5538913
650	234			31009.781	436678.08	0.54	0.552231
655	234			31212.912	440037.15	0.54	0.5505639
660	234			31416.043	443396.21	0.54	0.5488899
665	234			31619.174	446755.27	0.54	0.5472089
670	234			31822.305	450114.33	0.54	0.5455211
675	234			32025.436	453473.4	0.54	0.5438262
680	234			32228.567	456832.46	0.54	0.5421243
685	234	0.21	0.33	32431.698	460191.52	0.54	0.5404155
690	234			32634.829	463550.58	0.54	0.5386995
695	234			32837.96	466909.64	0.54	0.5369765
700	234			33041.091	470268.71	0.54	0.5352463
705	234			33244.222	473627.77	0.54	0.533509
710	234			33447.354	476986.83	0.54	0.5317645
715	234			33650.485	480345.89	0.54	0.5300128
720	234			33853.616	483704.96	0.54	0.5282539
725	234			34056.747	487064.02	0.54	0.5264876
730	234			34259.878	490423.08	0.54	0.5247141
735	234			34463.009	493782.14	0.54	0.5229333
740	234			34666.14	497141.2	0.54	0.5211451
745	234			34919.271	501400.27	0.54	0.5188672
750	234			35072.402	503859.33	0.54	0.5175465
755	234			35275.533	507218.39	0.54	0.5157361
760	234			35478.664	510577.45	0.54	0.5139182
765	234			35681.795	513936.51	0.54	0.5120927
770	234			35884.927	517295.58	0.54	0.5102597
775	234			36088.058	520654.64	0.54	0.5084192
780	234			36291.189	524013.7	0.54	0.506571
785	234			36494.32	527372.76	0.54	0.5047152
790	234			36697.451	530731.83	0.54	0.5028517
795	234			36900.582	534090.89	0.54	0.5009806
800	234			37103.713	537449.95	0.54	0.4991017
805	234			37306.844	540809.01	0.54	0.497215
810	234			37509.975	544168.07	0.54	0.4953205
815	234			37713.106	547527.14	0.54	0.4934182
820	234			37916.237	550886.2	0.54	0.4915081
825	234			38119.368	554245.26	0.54	0.4895901
830	234			38322.5	557604.32	0.54	0.4876641
835	234			38525.631	560963.39	0.54	0.4857302
840	234			38728.762	564322.45	0.54	0.4837882
845	234			38931.893	567681.51	0.54	0.4818383
850	234			39135.024	571040.57	0.54	0.4798803
855	234			39338.155	574399.63	0.54	0.4779142
860	234			39541.286	577758.7	0.54	0.47594

APPENDIX F

EXPECTED UTILITY COMPARISION AMONGST MACHINES

The third decision of the hearing aid example problem is the determination of the "cutoff" points of the ranges for each machine type / quantity combination. These results are presented in Table 5.10. The details of the comparision process are presented in this appendix.

Capacity	U_1Maxum	U_2Maxum	U_1Titan	U_1Prod+	U_2Prod+		U_Max	Machine
120	0.9550454	0.9129942	0.9124272	0.8780461	0.8755902		0.9550454	1Maxum
125	0.9550454	0.9129942	0.9124272	0.8780461	0.8755902		0.9550454	1Maxum
130	0.9529377	0.9111268	0.9098929	0.8761824	0.8725005		0.9529377	1Maxum
135	0.9507826	0.9092266	0.9073478	0.8743021	0.869394		0.9507826	1Maxum
140	0.9476174	0.9073483	0.9047747	0.8724157	0.8662707		0.9476174	1Maxum
145	0.9463987	0.9046446	0.9022081	0.8705234	0.8631305		0.9463987	1Maxum
150	0.9443531	0.9032913	0.8995965	0.8686249	0.8599732		0.9443531	1Maxum
155	0.942301	0.9015755	0.896956	0.8667205	0.8567988		0.942301	1Maxum
160	0.9392265	0.8998548	0.8943034	0.8648099	0.8536071		0.9392265	1Maxum
165	0.938126	0.8981073	0.8916385	0.8628932	0.8503982		0.938126	1Maxum
170	0.9361713	0.8963772	0.8889614	0.8609704	0.8471718		0.9361713	1Maxum
175	0.9342107	0.8937944	0.886272	0.8590414	0.8439279		0.9342107	1Maxum
180	0.9312167	0.8927464	0.8835702	0.8571063	0.8406664		0.9312167	1Maxum
185	0.9301953	0.8910852	0.8808559	0.855165	0.8373872		0.9301953	1Maxum
190	0.928322	0.8894393	0.8781292	0.8532175	0.8340901		0.928322	1Maxum
195	0.9264432	0.8877889	0.8742855	0.8512637	0.8307752		0.9264432	1Maxum
200	0.9234929	0.8852824	0.8716269	0.8493037	0.8274423		0.9234929	1Maxum
205	0.9223631	0.8843116	0.8690563	0.8473375	0.8240912		0.9223631	1Maxum
210	0.920723	0.8827404	0.8653678	0.8453649	0.820722		0.920723	1Maxum
215	0.9191041	0.8811652	0.8638818	0.8433861	0.8173344		0.9191041	1Maxum
220	0.9174555	0.8795692	0.8612777	0.8414009	0.8139284		0.9174555	1Maxum
225	0.9147386	0.8779861	0.8586624	0.8394093	0.8105039		0.9147386	1Maxum
230	0.9141156	0.8755413	0.8560357	0.8374114	0.8070608		0.9141156	1Maxum
235	0.9124983	0.8747124	0.8533977	0.8354071	0.803599		0.9124983	1Maxum
240	0.9108768	0.8732008	0.8507482	0.8333964	0.8001184		0.9108768	1Maxum
245	0.9081785	0.8716855	0.8480873	0.8313793	0.7957644		0.9081785	1Maxum
250	0.907561	0.8701515	0.8454148	0.8293556	0.7928524		0.907561	1Maxum
255	0.9060092	0.867782	0.8427308	0.8273256	0.7895009		0.9060092	1Maxum
260	0.9044535	0.8669268	0.8400351	0.825289	0.7861319		0.9044535	1Maxum
265	0.9018165	0.8656005	0.8373278	0.8232459	0.7827454		0.9018165	1Maxum
270	0.901292	0.8642578	0.8346087	0.8211962	0.7793412		0.901292	1Maxum
275	0.8997631	0.8629259	0.8307066	0.81914	0.7759193		0.8997631	1Maxum
280	0.8982517	0.8615778	0.8291352	0.8170772	0.7724795		0.8982517	1Maxum
285	0.8956089	0.8593682	0.8263807	0.8150078	0.7690218		0.8956089	1Maxum
290	0.8950539	0.8588065	0.8236141	0.8129318	0.765546		0.8950539	1Maxum
295	0.8935819	0.8574714	0.8208356	0.8108491	0.7620521		0.8935819	1Maxum
300	0.8921064	0.8561335	0.8180451	0.8087598	0.7585399		0.8921064	1Maxum
305	0.8895378	0.8547928	0.8140401	0.8066637	0.7550095		0.8895378	1Maxum
310	0.8890619	0.8525975	0.8124276	0.804561	0.7514606		0.8890619	1Maxum
315	0.8877247	0.8520238	0.8096006	0.8024515	0.7478932		0.8877247	1Maxum
320	0.886366	0.85074	0.8067614	0.8003352	0.7443071		0.886366	1Maxum
325	0.8850042	0.8494536	0.8039098	0.7982122	0.7407024		0.8850042	1Maxum
330	0.8825518	0.8481646	0.8010458	0.7960824	0.7370788		0.8825518	1Maxum
335	0.8822161	0.846873	0.7981694	0.7939457	0.7334363		0.8822161	1Maxum
340	0.8808771	0.8447214	0.7952806	0.7918022	0.7297747		0.8808771	1Maxum
345	0.8795351	0.8442647	0.7923792	0.7896518	0.7260941		0.8795351	1Maxum
350	0.8770962	0.8429959	0.7894652	0.7874946	0.7223942		0.8770962	1Maxum
355	0.8768049	0.8417245	0.7852825	0.7853304	0.718675		0.8768049	1Maxum
360	0.8754848	0.8404506	0.7835992	0.7831592	0.7149363		0.8754848	1Maxum
365	0.8741617	0.8383108	0.780647	0.7809812	0.7111781		0.8741617	1Maxum
370	0.8717356	0.8375395	0.7776821	0.7787961	0.7064385		0.8717356	1Maxum
375	0.8714839	0.8364066	0.7747043	0.776604	0.7036027		0.8714839	1Maxum
380	0.8701818	0.8352716	0.7717135	0.7744049	0.6997852		0.8701818	1Maxum
385	0.868877	0.8341346	0.7687098	0.7721987	0.6959478		0.868877	1Maxum
390	0.866463	0.8329955	0.765693	0.7699855	0.6920903		0.866463	1Maxum
395	0.8662465	0.8309946	0.7613624	0.7677652	0.6882127		0.8662465	1Maxum
400	0.8649618	0.8306675	0.7596201	0.7655377	0.6843148		0.8649618	1Maxum
405	0.8636745	0.829549	0.7565638	0.7633031	0.6803964		0.8636745	1Maxum
410	0.8612721	0.8284286	0.7534942	0.7610613	0.6764576		0.8612721	1Maxum
415	0.8609816	0.8273063	0.7504113	0.7588123	0.6724982		0.8609816	1Maxum
420	0.8597398	0.825317	0.747315	0.7565561	0.668518		0.8597398	1Maxum
425	0.8584955	0.8250319	0.7442052	0.7542927	0.664517		0.8584955	1Maxum
430	0.8572487	0.8239292	0.741082	0.752022	0.6604951		0.8572487	1Maxum
435	0.8548826	0.8228247	0.7365981	0.749744	0.6564521		0.8548826	1Maxum
440	0.8547943	0.8217183	0.7347946	0.7474587	0.6523879		0.8547943	1Maxum
445	0.853627	0.8206099	0.7316303	0.7451661	0.6483025		0.853627	1Maxum
450	0.8524574	0.8186284	0.7284524	0.7428661	0.6441957		0.8524574	1Maxum
455	0.8501721	0.8183146	0.7252606	0.7405587	0.6400673		0.8501721	1Maxum
460	0.8501161	0.8172486	0.7220549	0.7382439	0.6359173		0.8501161	1Maxum
465	0.8489653	0.8161809	0.7188352	0.7359217	0.6317456		0.8489653	1Maxum
470	0.8478123	0.8151114	0.7156016	0.733592	0.627552		0.8478123	1Maxum

Capacity	U_1Maxum	U_2Maxum	U_1Titan	U_1Prod+	U_2Prod+		U_Max	Machine
565	0.8264456	0.7953397	0.6514151	0.6878779	0.5795453		0.8264456	1Maxum
570	0.825366	0.7943923	0.6478877	0.6853938	0.577991		0.825366	1Maxum
575	0.8242846	0.7934435	0.6443449	0.6829017	0.5764303		0.8242846	1Maxum
580	0.8220587	0.7924934	0.6392571	0.6804016	0.5748632		0.8220587	1Maxum
585	0.8220025	0.7915418	0.6372129	0.6778934	0.5732896		0.8220025	1Maxum
590	0.8210038	0.7897032	0.6336235	0.6753772	0.5717095		0.8210038	1Maxum
595	0.8200035	0.7896082	0.6284686	0.6728529	0.5701228		0.8200035	1Maxum
600	0.8178676	0.7886719	0.6263977	0.6703204	0.5685296		0.8178676	1Maxum
605	0.8179981	0.7877343	0.6227612	0.6677798	0.5669298		0.8179981	1Maxum
610	0.8169929	0.7867953	0.6190052	0.665231	0.5653235		0.8169929	1Maxum
615	0.8159862	0.7849656	0.6150082	0.6644976	0.5637104		0.8159862	1Maxum
620	0.8138364	0.7848776	0.6108431	0.6639324	0.5616556		0.8138364	1Maxum
625	0.8139641	0.7839532	0.6067421	0.6631397	0.5604643		0.8139641	1Maxum
630	0.8129723	0.7830275	0.6025881	0.6623451	0.5588312		0.8129723	1Maxum
635	0.8119789	0.7821005	0.5982171	0.6615485	0.5571914		0.8119789	1Maxum
640	0.810984	0.7811722	0.5937512	0.66075	0.5555447		0.810984	1Maxum
645	0.8088408	0.7793485	0.5893776	0.6599495	0.5538913		0.8088408	1Maxum
650	0.8089794	0.7792833	0.5844801	0.659147	0.552231		0.8089794	1Maxum
655	0.8079989	0.778369	0.5796397	0.6583425	0.5505639		0.8079989	1Maxum
660	0.8070169	0.7774535	0.5743286	0.657536	0.5488899		0.8070169	1Maxum
665	0.8048832	0.7765368	0.5687122	0.6567276	0.5472089		0.8048832	1Maxum
670	0.805012	0.7747211	0.5622485	0.6559171	0.5455211		0.805012	1Maxum
675	0.8040439	0.7746583	0	0.6551046	0.5438262		0.8040439	1Maxum
680	0.8030742	0.7737551	0	0.6542902	0.5421243		0.8030742	1Maxum
685	0.8009496	0.7728506	0	0.6534737	0.5404155		0.8009496	1Maxum
690	0.8011304	0.7719449	0	0.6526552	0.5386995		0.8011304	1Maxum
695	0.8001563	0.771038	0	0.6518346	0.5369765		0.8001563	1Maxum
700	0.7991807	0.7692275	0	0.6510121	0.5352463		0.7991807	1Maxum
705	0.797043	0.7691823	0	0.6501875	0.533509		0.797043	1Maxum
710	0.797234	0.7682884	0	0.6493608	0.5317645		0.797234	1Maxum
715	0.7962719	0.7673933	0	0.6485322	0.5300128		0.7962719	1Maxum
720	0.7953083	0.7664969	0	0.6477014	0.5282539		0.7953083	1Maxum
725	0.7931795	0.7646938	0	0.6468687	0.5264876		0.7931795	1Maxum
730	0.7933769	0.7646613	0	0.6460338	0.5247141		0.7933769	1Maxum
735	0.7924089	0.7638322	0	0.6451969	0.5229333		0.7924089	1Maxum
740	0.7914396	0.7630022	0	0.644358	0.5211451		0.7914396	1Maxum
745	0.7904687	0.7621711	0	0.6435169	0.5188672		0.7904687	1Maxum
750	0.7883237	0.7613389	0	0.6426738	0.5175465		0.7883237	1Maxum
755	0.7883989	0.7596048	0	0.6418286	0.5157361		0.7883989	1Maxum
760	0.7875128	0.7596715	0	0.6409813	0.5139182		0.7875128	1Maxum
765	0.7866255	0.7588362	0	0.6401319	0.5120927		0.7866255	1Maxum
770	0.784578	0.7579998	0	0.6392804	0.5102597		0.784578	1Maxum
775	0.7848471	0.7571624	0	0.6384268	0.5084192		0.7848471	1Maxum
780	0.783956	0.7554175	0	0.6375711	0.506571		0.783956	1Maxum
785	0.7830637	0.7554865	0	0.6367133	0.5047152		0.7830637	1Maxum
790	0.7810047	0.7546617	0	0.6358533	0.5028517		0.7810047	1Maxum
795	0.781259	0.7538358	0	0.6349912	0.5009806		0.781259	1Maxum
800	0.7803793	0.753009	0	0.634127	0.4991017		0.7803793	1Maxum
805	0.7794983	0.752181	0	0.6332607	0.497215		0.7794983	1Maxum
810	0.7774482	0.7504418	0	0.6323922	0.4953205		0.7774482	1Maxum
815	0.7777328	0.7505132	0	0.6315216	0.4934182		0.7777328	1Maxum
820	0.7768483	0.7496973	0	0.6306488	0.4915081		0.7768483	1Maxum
825	0.7759625	0.7488805	0	0.6297739	0.4895901		0.7759625	1Maxum
830	0.7739011	0.7480627	0	0.6288968	0.4876641		0.7739011	1Maxum
835	0.7741873	0.7472439	0	0.6280175	0.4857302		0.7741873	1Maxum
840	0.7732979	0.7455099	0	0.627136	0.4837882		0.7732979	1Maxum
845	0.7724072	0.745582	0	0.6262524	0.4818383		0.7724072	1Maxum
850	0.7715154	0.7447749	0	0.6253666	0.4798803		0.7715154	1Maxum
855	0.7694397	0.7439667	0	0.6244786	0.4779142		0.7694397	1Maxum
860	0.769728	0.7431576	0	0.6235884	0.47594		0.769728	1Maxum
865	0.7688324	0.7414307	0	0.622696	0.4739576		0.7688324	1Maxum
870	0.7679357	0.7414878	0	0.6218014	0		0.7679357	1Maxum
875	0.7658485	0.7406899	0	0.6209046	0		0.7658485	1Maxum
880	0.7661384	0.739891	0	0.6200056	0		0.7661384	1Maxum
885	0.765238	0.7390912	0	0.6191043	0		0.765238	1Maxum
890	0.7643363	0.7382905	0	0.6182008	0		0.7643363	1Maxum
895	0.7622375	0.7365682	0	0.6172951	0		0.7622375	1Maxum
900	0.7625292	0.7366861	0	0.6163872	0		0.7625292	1Maxum
905	0.7616238	0.7358825	0	0.615477	0		0.7616238	1Maxum
910	0.7607171	0.735078	0	0.6145645	0		0.7607171	1Maxum
915	0.7586068	0.7342725	0	0.6136498	0		0.7586068	1Maxum

REFERENCES

Anderson, D. M., 1997, *Agile Product Develpment for Mass Customization*, Irwin Professional Publishing, Chicago.

Bednar, H. H., 1986, *Pressure Vessel Design Handbook*, Van Nostrand Reinhold, NY.

Bejan, A., 1996, "Street Network Theory of Organization in Nature," *Journal of Advanced Transportation,* Vol. 255, No. 7, pp. 85 - 107.

Bejan, A., 1997, *Advanced Engineering Thermodynamics*, John Wiley & Sons, New York.

Bejan, A., 2000, *Shape and Structure: From Engineering to Nature*, Cambridge University Press.

Belegundu, A. D. and T. R. Chandrupatla, 1999, *Optimization Concepts and Applications in Engineering*, Prentic-Hall, Inc., Upper Saddle River, NJ.

Carone, M. J., C. B. Williams, J. K. Allen and F. Mistree, 2003, "An Application of Constructal Theory in the Multi-Objective Design of Product Platforms," *Proceedings of the 2003 ASME Design Engineering Technical Conferences*, Chicago, Il. DETC2003/DTM-48667.

Engineous, July 2003, "i-SIGHT Software", http://www.engineous.com.

Ericsson, A. and G. Erixon, 1999, *Controlling Design Variants: Modular Platforms*, ASME Press, Dearborn, MI.

Fernández, M. G., 2002, *On Decision Support for Distributed Collaborative Design and Manufacturing*, M.S. Thesis, G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA.

Fernández, M. G., C. C. Seepersad, D. W. Rosen, J. K. Allen and F. Mistree, 2001, "Utility-Based Decision, Support for Selection in Engineering Design," *13th Internation Conference on Design Theory and Methodology*, Pittsburgh, PA, DETC2001/DAC-21106.

General Motors Corporation, May, 2003, "Important GM Dates: 1920 - 1929", http://media.gm.com/corpcom/history/gmdat20s.htm.

Hayter, A. J., 2002, *Probability and Statistics for Engineers and Scientists*, Thomson Learning, Pacific Grove, CA.

Hernandez, G., 2001, *Platform Design for Customizable Products as a Problem of Access in Geometric Space*, Ph.D. Dissertation, G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA.

Hernandez, G., J. K. Allen and F. Mistree, 2002, "Design of Hierarchic Platforms for Customizable Products," *Proceedings of the 2002 ASME Design Automation Conference*, Monteral, Canada, DETC2002/DAC-34095.

Hernandez, G., J. K. Allen and F. Mistree, 2003, "Design of Platforms for Customizable Products as a Problem of Access in a Geometric Space," *Journal of Engineering Optimization*, in press.

Hopp, W. J. and M. L. Spearman, 2001, *Factory Physics*, McGraw-Hill, Boston.

Jacobs, O. L. R., 1967, *An Introduction to Dynamic Programming: The Theory of Multistage Decision Processes*, Chapman and Hall, London.

Kalpakjian, S., 1997, *Manufacturing Processes for Engineering Materials*, Addison Wesley, New York.

Keeney, R. L. and H. Raiffa, 1976, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, John Wiley & Sons, New York.

Lehnerd, A. P., 1987a, "Revitalizing the Manufacture and Design of Mature Global Product," *Technology and Global Industry: Companies and Nations in the World Economy*, (H. Brooks, eds.), National Academy Press, Washington, D.C., pp. 49-64.

Lehnerd, A. P., 1987b, "Revitalizing the Manufacture and Design of Mature Global Product," *Technology and Global Industry: Companies and Nations in the World Economy (*Guile, B.R. and Brooks, H., eds.*)*.

Masters, M., 2002, "Direct Manufacturing of Custom-Made Hearing Instruments," *SME Rapid Prototyping Conference and Exhibition*, Cincinnati, OH.

Minderhoud, S., 2003, "Philips Centre for Industrial Technology," Atlanta, May 19.

Mistree, F., O. F. Hughes and B. A. Bras, 1993a, "The Compromise Decision Support Problem and the Adaptive Linear Programming Algorithm," *Structural Optimization: Status and Promise*, (*M. P. Kamat, eds.*), AIAA, Washington, D.C., pp. 247-286.

Mistree, F., W. F. Smith and B. A. Bras, 1993b, "A Decision-Based Approach to Concurrent Engineering," *Handbook of Concurrent Engineering*, (*H. R. Paresai and W. Sullivan, eds.*), Chapman & Hall, New York, pp. 127-158.

myb2o.com, February 2001, "The Production Paradigm", http://www.myb2o.com/myb2ous/MassCustomization/Tools/Process/10620.htm.

Nayak, R. U., W. Chen, J. K. Allen and F. Mistree, 1999, "A Variation-Based Methodology for Product Family Design," *Proceedings of the 2000 ASME Design Automation Conference*, Baltimore, MD., DETC2000/DAC-8876.

Nelson, S. A., M. B. Parkinson and P. Y. Papalambros, 2001, "Multicriteria Optimization in Product Platform Design," *Journal of Mechanical Design,* Vol. 123, No. 2, pp. 199-204.

Panchal, J. H., M. K. Chamberlain, D. W. Rosen, J. K. Allen and F. Mistree, 2002, "A Service Based Architecture for Information and Asset Utilization in Distributed Product Realization," *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, GA., AIAA-2002-5480.

Pederson, K., J. Emblemsvag, J. K. Allen and F. Mistree, 2000, "Validating Design Methods and Research - The Validation Square," *ASME Design Theory and Methodology Conference*, Baltimore, MA, DETC00/DTM-14579.

Pine, B. J., 1993, *Mass Customization: The New Frontier in Business Competition*, Harvard Business School Press, Boston.

Seepersad, C. C., 2001a, *The Utility-Based Compromise Decision Support Problem with Application in Product Platform Design*, M.S. Thesis, G. W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA.

Seepersad, C. C., 2001b, *The Utility-Based Compromise Decision Support Problem with Applications in Product Platform Design*, The G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA.

Seepersad, C. C., G. Hernandez and J. K. Allen, 2000, "A Quantitative Approach to Determining Product Platform Extent," *ASME Advances in Design Automation Conference*, Baltimore, MD, DETC2000/DAC-14288.

Seepersad, C. C., F. Mistree and J. K. Allen, 2002, "A Quantitative Approach for Designing Multiple Product Platforms for an Evolving Portfolio of Products," *ASME Design Engineering Technical Conferences, Advances in Design Automation*, Montreal, Canada, DETC2002/DAC-34096.

Siddique, Z. and D. W. Rosen, 2000, "Product Family Configuration Reasoning Using Discrete Design Spaces," *ASME Design Theory and Methodology Conference*, Baltimore, MD, DETC2000/DTM-14666.

Simon, H. A., 1996, *Sciences of the Artificial*, The MIT Press, Cambridge, MA.

Simon, H. A. and A. Ando, 1961, "Aggregation of Variables in Dynamic Systems," *Econometrica*, No. 29, pp. 111-138.

Simpson, T. W., 1999, *A Concept Exploration Method for Product Family Design*, Ph. D. Dissertation, G. W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta.

Simpson, T. W., W. Chen, J. K. Allen and F. Mistree, 1999, "Use of the Robust Concept Exploration Method to Facilitate the Design of a Family of Products," *Simultaneous Engineering: Methodologies and Applications*, (*U. Roy and J. M. Usher, eds.*), Chapman-Hall, New York, pp. 247-278.

Simpson, T. W., J. R. A. Maier and F. Mistree, 2001, "Product Platform Design: Method and Application," *Research in Engineering Design,* Vol. 13, pp. 2-22.

Stratasys, July 2003, "Stratasys FDM Technologies", http://www.stratasys.com.

Tseng, M. and J. Jiao, 1998, "Design for Mass Customization by Developing Product Family Architecture," *ASME Design for Manufacture Conference*, Atlanta, GA, DETC98/DFM-5717.

Williams, C. B., J. H. Panchal and D. W. Rosen, 2003, "A General Decision-Making Method for the Rapid Manufacturing of Customized Parts," *Proceedings of the 2003 ASME Computer and Information in Engineering Conference*, Chicago, Ill., DETC2003/CIE-48198.