Optimization of Autoclave Processing of Polymer Composites Using a GeneticAlgorithm

by

R.Mark Shead

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree of

MASTER OF SCIENCE

Department of Mechanical and Industrial Engineering

University of Manitoba

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION PAGE

Optimization of Autoclave Processing of Polymer Composites Using a Genetic Algorithm

BY

R. Mark Shead

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

ABSTRACT

Current industrial autoclave cure cycles used to process composites in the aerospace industry are conservative and costly. It is therefore desirable to optimize cure cycles based on industrial quality requirements to reduce costs. Process optimization requires several key components including a process model, an optimizing scheme, and an objective function. To address this problem, the University of Manitoba and its industrial partners have set out to develop the Advanced Process Control System (APCS). As a portion of this project, the primary objective of this thesis was to develop an objective function and interface with a genetic algorithm based path generator and a process model COMPRO, developed at The University of British Columbia, to develop the optimizer for autoclave process cycle optimization. Models run by the process model used Cytec-Fiberite 934 with Toray T300 fibres. As a secondary objective for the thesis, material characterization was performed for Hexcel F155 resin with Toho T300 carbon fibres for use in future work with APCS. Heat transfer and warpage predictions made by the process model were validated by experiments performed at Boeing Technology Canada – Winnipeg Division. This thesis discusses the effects of various versions of the objective function and the inputs required by the genetic algorithm. Based on the optimization results for various optimizer settings used in the study, the thesis concludes by providing the best optimizer configuration.

ACKNOWLEDGEMENTS

Thank you to my fellow students for their support and words of encouragement:

- Michael Hudek for his assistance with boundary conditions for the process model as well as all the support with the GA code.
- Madhava Koteshwara for assistance with material testing and supplying me with the material data for 934.
- Sue Lee for programming the fitness function routine.

Thank you to the industrial partners:

- Dr. Andrew Johnson for making himself completely available for assistance.
- Loren Hendrickson and Dwayne Johnson for help with the TMA and allowing me access to their facilities and their knowledge of composites.
- Rick Cole for help with laying out industrial requirements.
- Nezih Mrad for his support on all components of the APCS project.

I would also like to thank:

- Daniel Tse for supplying the majority of the code for the GA.
- The other members of MR&D at Boeing Technology - Winnipeg Division for their assistance with sample preparation.

Thank you to NSERC, NRC, Bristol Aerospace Ltd, Boeing Technology Canada – Winnipeg Division, and The University of Manitoba for financial assistance.

A special thanks to:

- My advisor, Dr. Raghavan Jayaraman, for providing me with the opportunity to be apart of such a diverse project.
- The people at the IAR in Ottawa for making our stay a wonderful experience.

A very special thanks to my family and friends for their words of encouragement and support over the past 2 years.

TABLE OF CONTENTS

Page Number

ABSTRACT i

ACKNOWLEDGEMENTS ii

TABLE OF CONTENTS iii

LIST OF FIGURES vii

LIST OF TABLES xi

CHAPTER 1 – INTRODUCTION AND BACKGROUND

1.0 Introduction..... 1

1.1 Advanced Process Control System (APCS)..... 2

1.1.1 Offline Optimization 3

1.1.2 On-line Optimization and Process Control..... 4

1.2 APCS Program Components..... 7

1.2.1 Process Model..... 7

1.2.1.1 Material Module..... 8

1.2.2 SIMCLAVE and ARS..... 8

1.2.3 Optimizer..... 11

1.2.3.1 Path Generator..... 11

1.2.3.2 Objective Function..... 11

1.2.4 Sensor Module..... 12

1.2.5 Control Module..... 12

1.2.6 Storage Module..... 13

1.2.7 On-Line Process Window..... 13

1.3 Scope of This Thesis..... 14

1.4 Thesis Outline..... 14

Page Number

CHAPTER 2 – LITERATURE REVIEW

2.0 Introduction..... 16

2.1. Optimizers..... 17

2.2 Expert Based Optimizers..... 18

2.3 Non-linear Programming Technique..... 21

2.4 Neural Networks..... 26

2.5 Heuristic Approach..... 29

2.6 Genetic Algorithm..... 34

2.7 Summary of Literature Review and Thesis Approach..... 37

CHAPTER 3 – OPTIMIZATION PROGRAM

3.0 Introduction..... 43

3.1 Structure of the Optimizer..... 44

3.1.1 Genetic Algorithm..... 44

3.1.2 Objective Function Module..... 47

3.1.2.1 Shape of Degree of Cure Function – $f(\alpha_{final})$..... 49

3.1.2.2 Shape of Maximum Temperature Function – f(T)..... 51

3.1.2.3 Shape of Warpage Function - f(warpage)..... 54

3.1.2.4 Shape of Time Function – f (t)..... 57

3.1.2.5 Shape of Temperature and Cure Gradient Functions - $f(\Delta T)$, $(\Delta \alpha)$ 61

3.1.3 Interfacing Modules..... 63

3.1.3.1 Process Model..... 63

3.1.3.2 Heat Transfer Predictions – SIMCLAVE and ARS..... 66

3.2 Operation of the Optimizer 68

3.2.1 Program Flow..... 68

3.2.2 Program Data Flow..... 70

3.2.3 Programming Notes..... 73

Page Number

CHAPTER 4 – EXPERIMENTAL AND SIMULATION DETAILS

4.0 Introduction..... 75

4.1 Material..... 75

4.2 Experimental Details..... 75

4.2.1 Cure Kinetics..... 76

4.2.2 Resin Degradation Temperature..... 82

4.2.3 Cure Shrinkage..... 84

4.2.4 Coefficient of Thermal Expansion (CTE)..... 85

4.2.5 Specific Heat Capacity..... 86

4.2.6 Viscosity..... 86

4.2.7 Gel Point..... 88

4.3 Manufacturing of Angle Laminates..... 88

4.4 Simulation Details..... 89

4.4.1 Process Model Inputs..... 89

4.4.2 SIMCLAVE Input Details..... 96

4.4.3 Optimizer Input Details..... 96

4.4.4 Optimization Test Matrix..... 99

4.4.4.1 Convergence..... 101

4.4.4.2 GA versus Micro-GA..... 101

4.4.4.3 Effect of Inputs in GAINPT.dat File..... 102

4.4.4.4 Influence of Input Cycles..... 103

4.4.4.5 Influence of Fitness Functions..... 104

4.4.4.6 Summary..... 111

CHAPTER 5 – RESULTS AND DISCUSSION

5.0 Introduction..... 114

5.1 Material Characterization Results..... 114

5.1.1 Cure Kinetics..... 114

5.1.2 Resin Degradation Temperature..... 117

5.1.3 Cure Shrinkage..... 117

Page Number

5.1.4 Coefficient of Thermal Expansion (CTE)..... 119
5.1.5 Specific Heat Capacity (Cp)..... 121
5.1.6 Viscosity..... 121
5.1.7 Gel Point..... 124
5.1.8 Summary of Material Properties..... 124
5.2 Simulations Results..... 124
5.2.1 Validation of Process Model..... 124
5.2.2 Optimization Results..... 128
5.2.2.1 Convergence..... 131
5.2.2.2 GA versus Micro-GA..... 134
5.2.2.3 Influence of Input Cycle..... 137
5.2.2.4 Effect of Inputs in GAINPT.dat File..... 141
5.2.2.5Influence of the Shape of Fitness Functions..... 144
5.2.2.6 Summary..... 149

CHAPTER 6 – CONCLUSIONS
6.0 Introduction..... 152
6.1 Material Data..... 153
6.2 Cure Cycle Optimization..... 156
6.2.1 Suggested Configuration for the Optimizer..... 156
6.3 Recommendations..... 158

REFERENCES..... 160

APPENDIX A
A.0 Introduction..... 166

APPENDIX B
B.0 Introduction..... 171

LIST OF FIGURES

Page Number

Figure 1.1: Functional Flow Chart of Off-Line Optimization 5

Figure 1.2: Functional Flow Chart for On-line Optimization 6

Figure 1.3: Schematic of Actual Autoclave Response versus Autoclave Set Points 10

Figure 2.1 : Diagram of Simple Neural Network Structure 28

Figure 2.2: Flow Chart of a Genetic Algorithm 35

Figure 3.1: Functional Flow Chart of Off-Line Optimization Program 45

Figure 3.2: Flow Chart of a Micro-GA 46

Figure 3.3: Shape of $f(\alpha_{final})$ versus Final Degree of Cure 50

Figure: 3.4: Fitness versus Temperature in Part During Cure 50

Figure 3.5 : Critical Distance "x" on Angle Composite Part 55

Figure 3.6 : General Shape of Objective Function versus Warpage (Distance Between Two Points) 55

Figure 3.7: Shape of Time Fitness Function – f(t) 58

Figure 3.8 : Fitness Function for Temperature and Cure Gradient 58

Figure 3.9: Schematic of Optimization Program Order of Operation 69

Figure 3.10: Schematic of Data Flow in Optimization Program 71

Figure 4.1: Total Heat of Reaction (Area Under Heat Flow vs. Time Plot) 79

Figure 4.2: Plot of Initial Isothermal Run, Re-Isothermal Run and Final Curves 79

Figure 4.3: Example Plot of an Isothermal Run at 100°C – Plot of Measured and Fitted $d\alpha/dt$ versus α 83

Figure 4.4: Plot of TMA Raw Data – Dimension Change (μm) versus time (min) 83

Figure 4.5: Tool "C" – Tool/Part used in Optimization Study 90

Figure 4.6: Picture of "Tool C" 90

Figure 4.7: Autoclave Loading Arrangements for Angle Laminates 91

Figure 4.8: Boundary Condition Numbers on the Finite Element Mesh 91

Figure 4.9: Cycle 1 Used for Warpage Study 94

Page Number

Figure 4.10: Cycle 2 Used in Warpage Study 95

Figure 4.11: Plot of Manufacturer's Recommended Cure Cycle 105
(for Cytec-Fiberite 934 Resin) and High Ramp Rate Cycle

Figure 4.12: Shape of $f(\alpha_{final})$ versus Final Degree of Cure – Version 2 108

Figure 4.13: Shape of $f(\alpha_{final})$ versus Final Degree of Cure – Version 3 108

Figure 4.14: Shape of f(t) versus Cycle Time – Version 2 110

Figure 5.1: Plot of Heat Flow versus Time for a Dynamic Scan of 2°C /min 116

Figure 5.2: Plot of α versus $d\alpha/dt$ for Isothermal Run at 110°C 116

Figure 5.3: Resin Degradation Temperature of Hexcel F155 Resin 118

Figure 5.4: Percent Cure Shrinkage versus Degree of Cure 118

Figure 5.5: Plot of Experimental and Predicted Strain versus Temperature Data 120

Figure 5.6: Plot of Experimental and Predicted Cp versus Temperature 120

Figure 5.7: Plot of Viscosity versus Temperature for all Dynamic Scans 122

Figure 5.8: Plot of Experimental and Predicted Values for Viscosity 122
versus α for a ramp rate of 2°C/min

Figure 5.9: Plot of Elastic and Viscous Modulus versus Temperature 125

Figure 5.10: Alpha versus Temperature for a Dynamic Scan of 2°C/min. 125
(Showing that gelation occurs at alpha =0.35)

Figure 5.11: Plot of Actual and Predicted Part Temperature versus Time 129
for "Tool C" Using Cycle 1 (from Figure 4.9) at 45 psi

Figure 5.12: Plot of Actual and Predicted Part Temperature versus Time 129
for "Tool C" Using Cycle 2 (from Figure 4.10) at 85 psi

Figure 5.13: Plot of Fitness vs. Number of Populations for Runs 1,3 and 4 132

Figure 5.14: Optimal Cycles for Runs 1-4 versus Manufacturer Cycle 132

Figure 5.15: Plot of Fitness versus Number of Populations for Runs 3 and 4 135

Figure 5.16: Plot of Fitness versus Number of Populations for Runs 7 and 8 135

Figure 5.17: Plot of Fitness versus Number of Populations for Runs 9 and 10 136

Figure 5.18: Plot of Fitness versus Number of Populations for Runs 11 and 12 136

Figure 5.19: Optimal Cycles With and Without Injected Cycles 139

Page Number

Figure 5.20: Fitness Plot for Runs 4, 6 and 8 139

Figure 5.21: Fitness Plot for Runs 12, 16 and 17 143

Figure 5.22: Optimal Cycles for Runs 12, 16 and 17 143

Figure 5.23: Fitness versus Number of Populations for Run 8, 12 and 13 145

Figure 5.24: Optimal Cure Cycles for Run 8, 12 and 13 145

Figure 5.25: Fitness versus Number of Populations for Run 12-15 148

Figure 5.26: Optimal Cure Cycles for Runs 12-15 148

Figure 5.27: Optimal Cure Cycles fro Runs 14 and 18 151

Figure A.1: Plot of α versus $d\alpha/dt$ for Isothermal Run at 90°C 167

Figure A.2: Plot of α versus $d\alpha/dt$ for Isothermal Run at 100°C 167

Figure A.3: Plot of α versus $d\alpha/dt$ for Isothermal Run at 110°C 168

Figure A.4: Plot of α versus $d\alpha/dt$ for Isothermal Run at 120°C 168

Figure A.5: Plot of α versus $d\alpha/dt$ for Isothermal Run at 130°C 169

Figure A.6: Plot of Experimental and Predicted Viscosity versus Degree of Cure for a Dynamic Scan of 2°C/min 169

Figure A.7: Plot of Experimental and Predicted Viscosity versus Degree of Cure for a Dynamic Scan of 3°C/min 170

Figure A.8: Plot of Experimental and Predicted Viscosity versus Degree of Cure for a Dynamic Scan of 5°C/min 170

Figure B1: Manufacturer's Recommended Cure Cycle 177

Figure B.2: Plot of High Ramp Rate Cycle 177

Figure B.3: Plot of Optimal Cycle from Run 1 178

Figure B.4: Plot of Fitness versus Number of Populations from Run 1 178

Figure B.5: Plot of Optimal Cycle from Run 2 179

Figure B.6: Plot of Fitness versus Number of Populations for Run 2 179

Figure B.7: Plot of Optimal Cycle from Run 3 180

Figure B.8: Plot of Fitness versus Number of Populations for Run 3 180

Figure B.9: Plot of Optimal Cycle from Run 4 181

Figure B.10: Plot of Fitness versus Number of Populations for Run 4 181

Page Number

Figure B.11: Plot of Optimal Cycle from Run 5 182

Figure B.12: Plot of Fitness versus Number of Populations for Run 5 182

Figure B.13: Plot of Optimal Cycle from Run 6 183

Figure B.14: Plot of Fitness versus Number of Populations for Run 6 183

Figure B.15: Plot of Optimal Cycle from Run 7 184

Figure B.16: Plot of Fitness versus Number of Populations for Run 7 184

Figure B.17: Plot of Optimal Cycle from Run 8 185

Figure B.18: Plot of Fitness versus Number of Populations for Run 8 185

Figure B.19: Plot of Optimal Cycle from Run 9 186

Figure B.20: Plot of Fitness versus Number of Populations for Run 9 186

Figure B.21: Plot of Optimal Cycle from Run 10 187

Figure B.22: Plot of Fitness versus Number of Populations for Run 10 187

Figure B.23: Plot of Optimal Cycle from Run 11 188

Figure B.24: Plot of Fitness versus Number of Populations for Run 11 188

Figure B.25: Plot of Optimal Cycle from Run 12 189

Figure B.26: Plot of Fitness versus Number of Populations for Run 12 189

Figure B.27: Plot of Optimal Cycle from Run 13 190

Figure B.28: Plot of Fitness versus Number of Populations for Run 13 190

Figure B.29: Plot of Optimal Cycle from Run 14 191

Figure B.30: Plot of Fitness versus Number of Populations for Run 14 191

Figure B.31: Plot of Optimal Cycle from Run 15 192

Figure B.32: Plot of Fitness versus Number of Populations for Run 15 192

Figure B.33: Plot of Optimal Cycle from Run 16 193

Figure B.34: Plot of Fitness versus Number of Populations for Run 16 193

Figure B.35: Plot of Optimal Cycle from Run 17 194

Figure B.36: Plot of Fitness versus Number of Populations for Run 17 194

Figure B.75: Plot of Optimal Cycle from Run 18 195

Figure B.38: Plot of Fitness versus Number of Populations for Run 18 195

LIST OF TABLES

	Page Number
Table 2.1 : Relationship Between Aerospace Industry Standard Composite Quality Measures and APCS	41
Table 4.1: Material Properties for Cytec-Fiberite 934 Resin and Toray T 300 Fibers	93
Table 4.2: Boundary Conditions on "Tool C"	94
Table 4.3: Measured and Predicted Springback Angles for "Tool C"	97
Table 4.4: Effective "C" Values for Convective Boundaries for Finite Element Model of "Tool C"	97
Table 4.5: Sample of QAMeas.txt file	98
Table 4.6: Sample GAINPT.dat File	100
Table 4.7: Settings and Objectives for Optimization Runs	112
Table 5.1: Summary of Material Properties for F155 with Toho T300 Fibers	126
Table 5.2: Optimal Cycle Times for Runs 5 to 12	138
Table 5.3: Optimization Data for Run 4, 6 and 8	138
Table 5.4: Optimization Data for Run 8, 12 and 13	146
Table 6.1: Summary of Material Properties for Hexcel F155 with Toho T300 Fibers	153
Table 6.2: Suggested Configuration for GAINPT.dat	157
Table B.1: Summary of Inputs used for Optimization Runs	173
Table B.2: Summary of Optimization Results	175
Table B.3: Summary of Results – Temperature and Cure Gradients	176

CHAPTER 1

Introduction and Background

1.0 Introduction

Composite materials have proven to be attractive for use primarily in secondary structures within the aerospace industry. In certain applications, composites offer superior mechanical properties when compared to traditional metallic components. In general, a composite is defined to be a material that is comprised of two or more dissimilar components. In the case of aerospace composites, these dissimilar components are the fibers and the matrix. Composites used in aircrafts, outside of engine components, typically use polymer resins as the matrix and are referred to as Polymer Matrix Composites (PMC).

Composite parts used in aircraft, vary greatly in size and shape and are typically cured in batch loads in autoclaves. The cure cycles differ depending on the matrix resin. However, in order to accommodate components made with different materials with similar cure cycles, within the same batch load, manufacturers use very conservative cure cycles. The conservative cycles ensure that parts are completely cured regardless of location within an autoclave and therefore have the required properties. It is therefore desirable to reduce cure cycle time, while maintaining part quality in order to improve efficiency, production flow and reduce production costs and the capital investment

required for large equipment such as autoclaves. As well as cycle time reduction, other factors such as the reduction of warpage and stress in a component could be important to industry as it could lead to better tool design and reduce parts dimensional variation from batch to batch.

In order to achieve a reduction of cure cycle time, warpage and residual stress , a software optimizer based upon sound understanding of the entire curing process is required. Much of the work to date use simple control systems or simple one dimensional process models (1-6). There is still a need for a more advanced optimizer and control system. For this reason, the University of Manitoba, the National Research Council, Bristol Aerospace Ltd., and Boeing Technology Canada Ltd, have started research on the Advanced Process Control System (APCS) of which this thesis is a portion of.

1.1 Advanced Process Control System (APCS)

The APCS optimizes cycles offline and then controls the autoclave online while monitoring sensors to ensure that the part is following the cure cycle. If the measured parameters deviate from the off-line optimized path, the APCS will re-optimize the process cycle such that final product quality is within specifications. The APCS requires several key modules and technologies in order to work. These components are as follows.

a) Reliable Process Model

- Material Property Data

b) SIMCLAVE and ARS (Autoclave Respose Simulator)

c) Optimizer

- Path Generation Module
- Objective Function Module

d) Sensor Module (Advanced Sensors)

e) Control Module

f) Storage Module

g) On-line Process Window

Components "a" through "c" are required for off-line optimization while component "d" through "g" are additionally required for on-line optimization.

1.1.1 Offline Optimization

A flow chart describing the function of the optimizer is given in Figure 1.1. The first step in off-line optimization is to setup the finite element model of the part using the process model by setting the mesh, material type, material lay-up, etc. Following this, SIMCLAVE is run to predict thermal boundary conditions based on orientation and location of the part within the autoclave. Subsequently, the actual optimization program begins.

The path generation module generates possible cure cycles and sends the cycles one at a time to the Objective Function Module (OFM). ARS (Autoclave Response

Simulator) predicts the actual air temperature in the autoclave using these cycles and outputs the air temperature in a format readable by the process model. The process model is run with this modified cure cycle and the results are transferred back to the OFM. The Objective Function Module uses selected data from the process model results to evaluate an objective function equation and assign a fitness for that cure cycle. This fitness value is evaluated against user-defined criterion to verify if that cure cycle is the optimal cycle. If the cycle is not optimal, the path generator then passes the next cure cycle to the Objective Function Module and the cycle continues until an optimal cycle is found.

1.1.2 On-line Optimization and Process Control

A functional flow chart is given in Figure 1.2. The off-line optimized cure cycle will be used to manufacture the composite part. The cure will be monitored on-line via sensors to ensure that the cure takes place as predicted. Sensing any deviation from the on-line process window, discussed in a section 1.2.7, the remaining portion of the cycle will be re-optimized on-line.

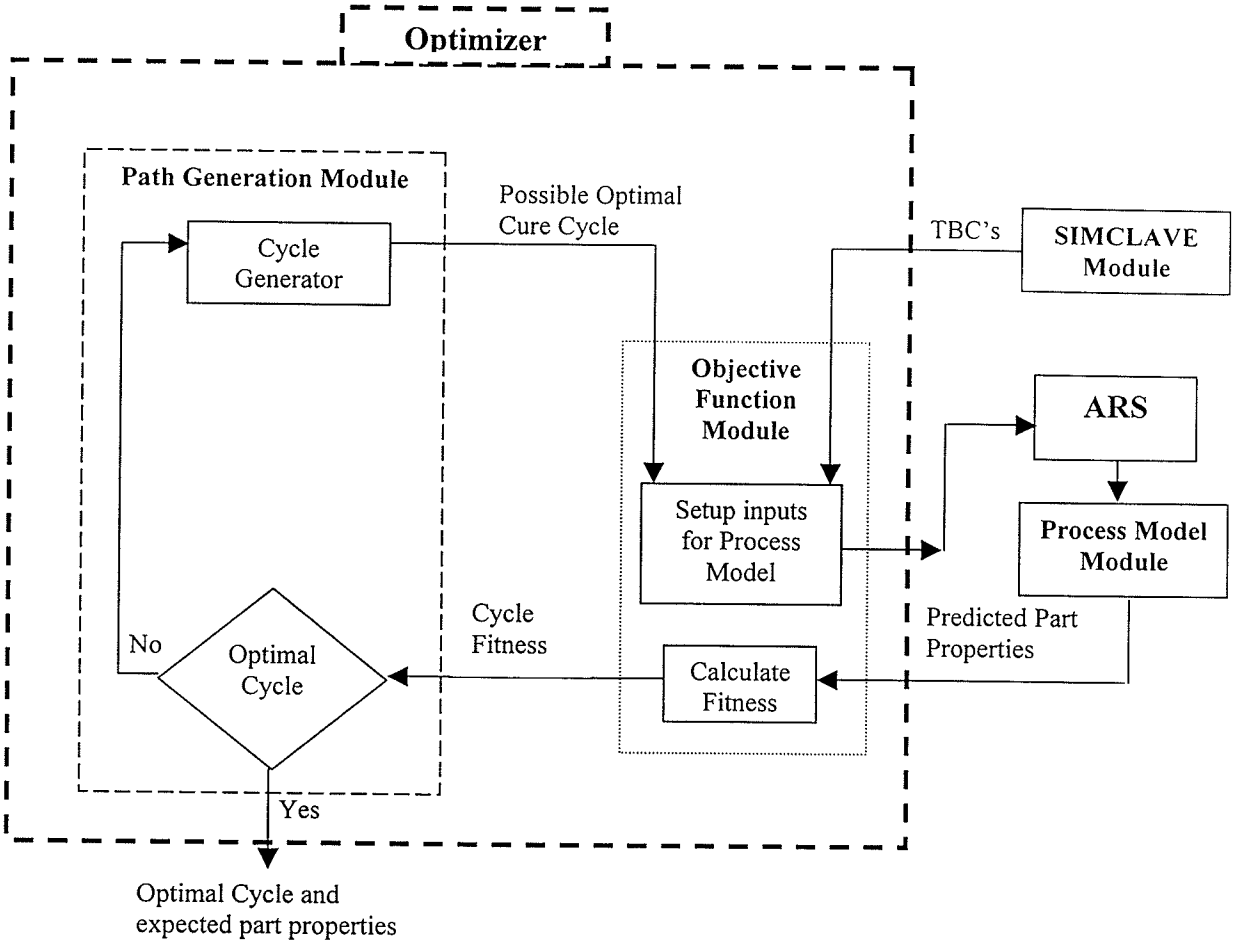

Figure 1.1: Functional Flow Chart of Off-Line Optimization

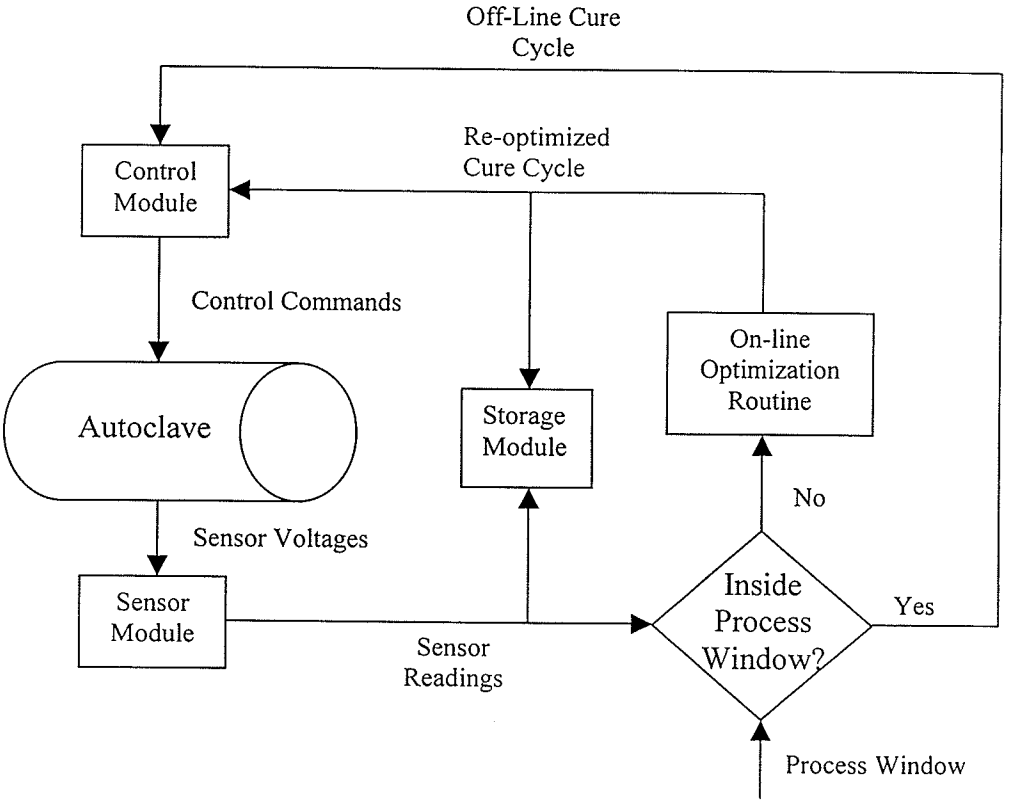

Figure 1.2: Functional Flow Chart for On-line Optimization

1.2 APCS Program Components

In the following sections, various modules of the APCS are discussed briefly.

1.2.1 Process Model

The purpose of the process model is to predict parameters such as part temperature, degree of cure, warpage, etc. in a part during the curing process. An accurate process model is of paramount importance for the proper and efficient operation of the APCS. Such a process model is available at the University of Manitoba. However, this model is ANSYS based and not portable at this time. Another process model, COMPRO developed at The University of British Columbia was used in this thesis. Details of COMPRO can be found in work published by Johnson (7).

Process models, such as COMPRO, have three main modules. The thermochemical module predicts the temperature and cure within the part, which are subsequently used by the other two modules. The second module is the flow module that predicts the resin flow, fiber volume fraction and final part thickness within the part. The last module is the stress module that calculates the build-up of residual stresses and part deformation during processing. This module is also capable of simulating tool removal from the part and therefore can predict final warpage of the part.

Accurate prediction by the process model requires reliable inputs to the process model. Inputs can be divided into two general areas – thermal boundary conditions and material property data.

1.2.1.1 Material Module

Each of the three modules in a process model, mentioned in section 1.2.1, require different material information. The thermochemical module requires material thermal properties such as thermal conductivity, specific heat capacity and resin cure kinetics data. The flow modules requires data such as viscosity as a function of cure. The third module requires material information such as coefficient of thermal expansion and modulus data.

Typically, the material properties mentioned above are determined experimentally or are taken from literature. This material data is then used to predict variables such as the degree of cure within the part, the temperature distribution, the build up of residual stresses, the warping of the part, etc.

1.2.2 SIMCLAVE and ARS

Heat transferred into the part determines the part temperature, which influences the rate of cure and the cure-dependent composite properties. Accurate prediction of temperature within the part is therefore paramount. In order to achieve this, accurate

heat transfer boundary conditions are required to be entered into the process model for use by the thermochemical module that was discussed above. It has been shown that there are many factors that influence thermal boundary conditions such as position and orientation of the part within the autoclave (8). In order to input accurate heat transfer boundary conditions to the process model, a program called SIMCLAVE has been developed by Michael Hudek at the University of Manitoba (9). SIMCLAVE is able to apply accurate heat transfer boundary conditions based on user inputs of position and location within the autoclave. The heat transfer boundary conditions are based on mathematical models developed from extensive experimentation. It is hoped that such a program will reduce the chances that a part will deviate from the predicted cure path. This will reduce the need for on-line optimization. More details are available in Master's Thesis by Michael Hudek at The University of Manitoba (9).

Air temperature is as important the heat transfer coefficient as the heat transfer into the part is dependant on the air temperature and the part temperature. As seen in Figure 1.3, autoclaves require time to reach the desired set points. Therefore using set points to determine actual air temperature can introduce error into a model. ARS (Autoclave Response Simulator), is being developed to translate the autoclave control set points into the actual response of the autoclave. This delivers more accurate air temperature predictions and therefore better simulates the heat transfer into the part.

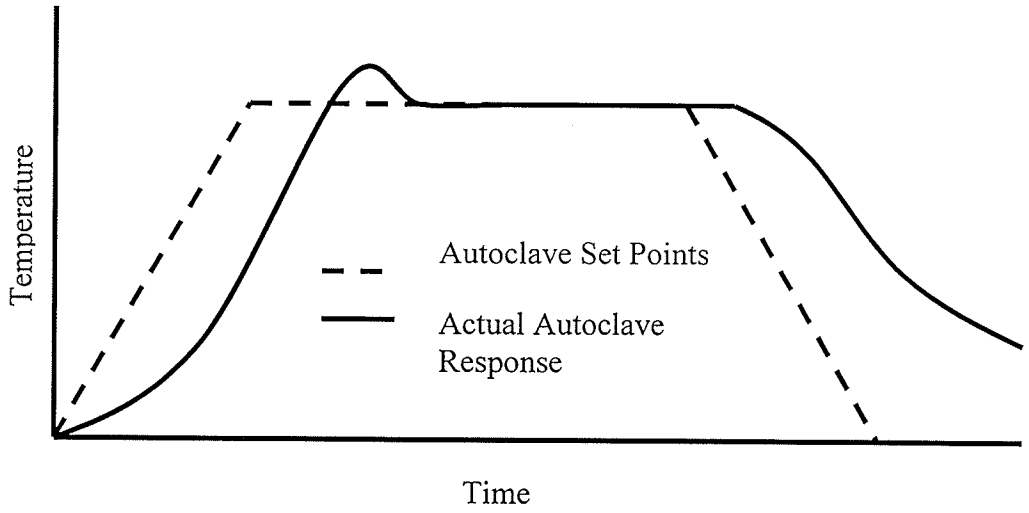

Figure 1.3: Schematic of Actual Autoclave Response versus Autoclave Set Points

1.2.3 Optimizer

The optimizer consists of two components, a path generator module (PGM) and an Objective Function Module (OFM). These are discussed in the following sections.

1.2.3.1 Path Generator

Various methods used in the past include quadratic programming, heuristic techniques and neural networks. Each of these systems are discussed in Chapter 2 in the literature review. A Genetic Algorithm (GA) approach has been chosen for use in APCS and in this thesis.

The GA simulates natural selection by generating a population of cycles that are each assigned a fitness value via the Objective Function Module (OFM). The GA selects cycles with the highest fitness and rejects those with poor fitness. Chosen cycles from the first population are crossed-over and mutated into a new generation of cycles. The GA continues through this cycle until an optimal cycle is reached based on a criterion.

1.2.3.2 Objective Function

A general objective equation is of the form:

$$fitness = A * f(x) + B * f(y) + C * f(z) + ... \qquad (1.1)$$

Where x, y, z are quality parameters and A, B, C are all weighting factors.

The quality parameters can be defined according to the desired quality to be optimized such as temperature, cure, etc. The objective function module collects values for the quality parameters from the output files of the process model and uses that data in equation 1.1 to determine the fitness for a given cure cycle. This fitness value is subsequently used by the path generation module (PGM) to rank cure cycles in order to generate the next set of population members. The objective function is setup so that the genetic algorithm drives towards the optimal solutions.

1.2.4 Sensor Module

Advanced sensors for properties such as cure and resin modulus will be used on-line to ensure that the process is following the optimal path predicted by the process model. Sensing deviation from the predicted path, this module will activate the on-line optimizer. The optimizer will re-optimize based on the sensor information and current autoclave conditions. This ensures that a part has the best opportunity to meet all the quality parameters.

1.2.5 Control Module

The Control Module oversees the process by collecting the sensor information and comparing it to the predicted information. The control module will also send commands to control the autoclave. If the sensor data shows deviation from the predicted

path, this module will activate the on-line optimizer and will send commands to the autoclave controller based on a re-optimized cure cycle.

1.2.6 Storage Module

The Storage Module stores sensor data during the cure as well as any re-optimized cure cycle. The purpose of the tracking is not only for trouble-shooting and inspection but validation and quality control purposes.

1.2.7 On-Line Process Window

Current manufacturer's cure cycles provide a range or window within which the composite will fully cure. For example, the manufacturer's recommended cure cycle for Hexcel F155 resin is as follows (10):

1. ramp to 127 °C at 1-4 °C/min
2. hold at 127 °C (+/- 6 °C) for 90 minutes (+15min –0 min)
3. ramp down to 30 °C at 3°C/minute (maximum).

This cure cycle provides a range or window in which the composite can be processed so that certain properties can be expected from the cycle. However, the optimizer will provide a single rigid cure cycle. For example, the optimizer could describe the cycle above as:

1. ramp to 127 °C at 2 °C/min
2. hold at 127 °C for 90 minutes
3. ramp down to 30 °C at 3°C/min.

An autoclave is not capable of following such a rigid setup and so a process window needs to be developed. The window is setup such that the part meets all the quality criteria no matter where in the window the cure takes place. If the cycle deviates from that window, then the on-line optimizer must be activated. The window must also account for the variability of material properties from batch to batch and the inability of the autoclave to follow a cure cycle exactly.

1.3 Scope of This Thesis

The scope of this thesis is to a) develop the optimizer module for the APCS, b) integrate the optimizer module with the other modules, c) demonstrate successfully the optimization capability of the optimizer and APCS, and d) characterization material property data for Hexcel F155 resin reinforced with Toho T300 carbon fiber.

1.4 Thesis Outline

A brief outline of the APCS and scope of this thesis has been provided in Chapter 1. Chapter 2 is a literature review of published literature on current control and

optimization systems. Chapter 3 provides a detailed discussion on the optimizer developed in this thesis as well as details on experiments and simulations. Chapter 4 presents and discusses experimental and simulation results. Finally, conclusions drawn from this work along with recommendations for future work are provided in Chapter 5.

CHAPTER 2

Literature Review

2.0 Introduction

In the past, several systems have been developed to control the curing of composite materials. These systems have met with varied success but have helped to lay the foundation for a project such as APCS (Advanced Process Control System). APCS has been developed based on the knowledge gained through intelligent control systems developed in the past. In addition, industrial input has also been sought and assimilated to create a practical control system.

Process Control Systems (PCS) developed in the past can be classified as expert-based (referred to as "expert systems" in this thesis) and model-based systems (referred to as "model systems" in this thesis). Expert systems use rules, derived from experiments and experience, to control the process. These rules specify the values and tolerances for the process parameters which cannot be violated during processing. Parameters usually include maximum temperature and minimum degree of cure which are monitored on-line to ensure that the cure takes place within the set of rules.

Model systems use a process model to predict parameters during a cure. The model systems use models that are based on scientific understanding of the relationship

between process parameters and quality indicators. The process models can be as simple as one dimensional which only taken into account a few parameters to multi-dimensional models that predict a much wider range of parameters. While there is a myriad of process control systems published in the literature, only those process systems used for controlling the composite curing are reviewed in subsequent sections.

A brief review of published research work on optimizers used in various process control systems is presented below. Objectives of this thesis, formulated based on this review, are presented subsequently.

2.1. Optimizers

Various types of optimizers used in the past include:

- Expert based systems
- Heuristic approach
- Non-linear optimization
- Neural Network
- Genetic algorithms

Each one of these are discussed below

2.2 Expert Based Optimizers

Ciriscioli, et al. (3) developed a rule based expert system program called SECURE. Expert systems rely on rules or cases developed based on an "expert's" experience and knowledge of important parameters. Rules for SECURE were developed to take into account: surface, midpoint, and autoclave temperatures, ionic conductivity to measure viscosity, autoclave pressure and part thickness to measure consolidation. The following is a list of conditions used to develop rules for SECURE (3):

1. The temperature at every point in the cure assembly must remain below a maximum value.
2. The temperature distribution across the laminate must remain below a maximum value.
3. Complete cure should be achieved in the shortest time period
4. Viscosity is low enough and for long enough for full compaction before the resin gels.
5. Void content should be a minimum
6. The residual stresses are small

From this list of criteria, it was decided that three parameters would be used to control the cure (3):

1. Midpoint temperature (T_m)

2. Temperature difference between midpoint and average surface temperature ($\Delta T = T_m - T_s$), where T_s, average surface temperature is the average of the top and bottom surface.

3. The rate of change of the midpoint temperature, T_{rate}

The rules developed from the criteria and the list of parameters mentioned above are (3):

IF

$$|\Delta T| < |\Delta T|_{max} \text{ and } T_{rate} \leq 0 \text{ and } T_m < T_{max} \text{ and } T_s < T_{max}$$

THEN heat autoclave

IF

$$|\Delta T| > |\Delta T|_{max} \text{ and } T_{rate} > 0 \text{ and } T_m \geq T_{max} \text{ and } T_s \geq T_{max}$$

THEN cool autoclave

The cure starts with an initial ramp rate. Following this, sensor data is taken and the controller takes over based upon the rules shown above. Results showed the controlled cure cycle was approximately 30 minutes shorter than the manufacturers cycle when curing a 200 ply laminate. However, the system required a large number of sensors to deliver the required information on the quality parameters to the control system which can make it impractical for a production environment.

Another example of an expert system was developed by Lee and Abrams (11). The system was called Qualitative Process Automation (QPA). The system used three thermocouples and one dielectric sensor for the input information used by the controller. Temperature data was used by the control system while dielectric data was used to determine when the part is fully cured. Rules were developed to control maximum temperature, temperature gradient and the rate of increase of temperature at the midpoint. It was reported that limiting the temperature gradient for thick composites (200 or more plies in the laminate) had the largest effect on controlling the exotherm. The control strategy seems to be effective for the given case studies. However, there was no mention of time savings in the report which is an important factor in an industrial context.

Trivisano et al. (12) presented a hybrid system. In this method, the optimal cure cycle was found while the initial ramp of the autoclave temperature was taking place. Using the heat transfer conditions recorded during the first 10 minutes of the heat up, the system optimized the cycle within next five minutes. Before the completion of the heat up of the autoclave was finished, the optimization routine provided the controller with the optimal cure cycle. Neither the process model nor the optimization method was discussed in detail in this paper. Based on this author's experience with the APCS, it can be concluded that the model and the optimizer must be extremely simple in order to operate in such a short period of time. Without sensors to confirm that full cure has taken place, this method is not reliable since the process model was too simple. It is also mentioned in the paper that the control system did not work well with thick parts.

The expert systems such as the ones mentioned above have inherent inflexibility. Rules are based upon an expert's knowledge are inherently limited to the systems for which the expert has experience. Therefore, these rules can usually be applied only to a limited number of cases and so many sets of rules would have to be developed and tested so that an expert system can have a wide enough range of application for use with the APCS.

Model based systems attempt to address problems inherent in expert systems by adding a more scientific approach to optimizations. Model systems discussed in the next sections include - non-linear programming, neural network, heuristics and genetic algorithms.

2.3 Non-linear Programming Technique

Non-linear programming is a group of mathematically based optimization algorithms. These algorithms consist of several modules that form the base of the algorithms. The three main modules are described in the following sections:

1. Solution Module

This module provides a search direction to follow that will hopefully lead to an optimal objective equation value and thus an optimized system. The objective equation is a relationship between the variables that are required to be optimized. When the equation value is minimized or maximized, as the case may be, the problem is said to be

optimized. The objective equation is user defined to describe the desired properties of an optimal system. The solution of the non-linear programming problem is iterative and thus what is seen as the correct search direction for one iteration may not be at a later point in the optimization. Therefore, at each iteration, the search direction must be reestablished by solving the quadratic programming problem (Biegler, 13). A sample quadratic programming problem equation is:

$$Q(x^I,B): \quad \text{Minimize } \nabla\phi(x^I)^T*d + \tfrac{1}{2}\, d^T*B*d \tag{2.1}$$

$$\text{Such that:} \quad g(x^I) + \nabla g(x^I)^T*d \le 0 \tag{2.2}$$

$$h(x^I) + \nabla h(x^I)^T*d = 0 \tag{2.3}$$

$$\text{for all:} \quad x_l \le x^I + d \le x_u \tag{2.4}$$

Where x^I is the I^{th} iteration x value, ϕ is the objective function, d is the search direction, B is the Hessian matrix, g is the inequality constraints, h is the equality constraints, x_l is the x value lower limit and x_u is the x value upper limit (Biegler, 13).

2. The Line Search Algorithm

The line search algorithm determines the appropriate step size in the search direction for a given iteration. Excessively large step sizes result in the optimizer oscillating around the function minimum and therefore never converge on an optimal solution. Solution steps that are too small may result in excessively long solution times. For these reasons, the line search algorithm is important in determining the efficiency of the optimizer. Usually, penalty functions are used to determine the optimal length of the

step based on the objective function value and through perturbation of the objective function to calculate the function gradients.

3. Scaling of the Algorithms

The quadratic programming problem must be scaled to fit the current iteration. The scale set can be very important to the success of the solution of the optimization problem because of the following reasons (Biegler, 13):

A. The initial Hessian matrix, B, in equation 2.1, should be based on the second derivative information of the ϕ, g and h. However, the second derivative of a function can be difficult to determine numerically. Therefore, the initial Hessian is usually set to an identity matrix and the scaling of the problem transforms the matrix into a reasonable approximation of the second derivative.

B. Inaccuracies in the gradients, round off error in the computer and the use of bad pivoting sequences in the matrices of the problem can disrupt the accuracy of the program.

For these reasons, the selection of a scaling algorithm for a optimization is problem dependent and experiments are often required to determine the appropriate values (Biegler, 13). The problem dependant nature of this part of non-linear programming makes it unattractive for the APCS just as this property makes expert systems unattractive.

Some authors report that the non-linear programming optimizers do not converge and get stuck in a loop (Beigler,13). Other authors, such as Vasantharajan (14), have reported successful optimization on a variety of programs. It should be noted that Vasantharajan used a large number of equality constraints as opposed to inequality constraints. Equality constraints help to decrease the search space and could be one explanation for Vasantharajan's success.

In terms of composite processing, non-linear programming techniques have been utilized in cure cycle optimization by Rai and Pitchumani (15). A simple process model was used to simulate the curing. This process model included a thermal module, a kinetics module and a void module. Parts were 0.5 inches and 1 inch thick and were constructed using 4 different prepregs. Little information is provided about the setup of the problem in term of the non-linear programming technique. However, the constraints used in the programs were described and are outlined below:

1. Criscioli et. al.(16), found that the stress in a composite was mostly dependant on the maximum temperature in the composite and therefore the temperature, T_{max} was not to exceed a critical value, $T_{critical}$. This critical value is material dependent property and can be expressed in the following inequality:

$$T_{max} - T_{critical} \leq 0 \qquad (2.5)$$

2. Another method of controlling excessive stresses is to limit the temperature gradient in the autoclave, dT/dt. This constraint can be written as:

$$\frac{dT}{dt}\max - \dot{T} \leq 0 \qquad (2.6)$$

In their study, the authors (15) set dT/dt_{max} to be from 4°C/min to 15°C/min.

3. In order to have a uniform cure, the thermal gradient in a cross-section of the part, ΔT_{max} was limited to a critical value, ΔT_{crit}. This is represented mathematically as:

$$\Delta T_{max} - \Delta T_{crit} \leq 0 \qquad (2.7)$$

The $\Delta T_{critical}$ was set from 10°C to 20°C for the case studies

4. To complete the cure, the minimum degree of cure in the component, α_{min} must be greater than a critical value, α_{crit} therefore:

$$\alpha_{crit} - \alpha_{min} \leq 0 \qquad (2.8)$$

Results of Rai and Pitchumani's (15) work claim up to a 57% decrease in processing time compared to the optimal cure cycle provided by heuristics (to be discussed in the next section). Unfortunately, there is no mention of the quality of the part after such a short cure cycle. The case studies considered by Rai and Pitchumani (15) took two to four hours of CPU time, while running on a Sparc 20 station, to provide a full cure cycle optimization. This solution time seems relatively quick when compared

to other possible methods. However, the quick solution could be attributed to the simplicity of the model being used. More complex models that include parameters such as flow, compaction, and warpage could be used. The inclusion of more parameters would increase solution time but could result in a more "optimal" cycle.

Non-linear programming techniques appear to be attractive in optimization due to the significant reduction in processing time. Unfortunately, non-linear optimization problems do not always converge. The study performed by Rai and Pitchumani (15) may work effectively if a more detailed model and restrictions are added to the optimizer. That said, the optimizer may not converge with the increased complexity of the model. Unfortunately, the unreliability and problem dependant nature of the optimizer makes non-linear programming unattractive for the APCS.

2.4 Neural Networks

Neural networks act essentially as simplified models. As seen in Figure 2.1, a neural network builds connections between input variables and the variables to be predicted. Intermediate connections in the hidden layers do not have to be known by the user as the neural network attempts to find a relationship for these connections. For composite processing, a neural network must be trained using a process model on the specific case for which the neural network is to be used. As an example, input variables might be sensor data such as temperature, pressure and viscosity while outputs could include temperature, compaction and degree of cure. After running a series of scenarios

in a process model, the input and output data is fed into the neural network which attempts to make connections between the two.

Once properly trained, neural networks are extremely fast and can operate in two possible modes –forward or inverse. In "forward" mode, input information is used to predict a future outcome. In "inverse" mode, the final desired values are used as input and the neural network determines what the present input conditions must be to achieve the desired effects.

An example of a neural network used in composite processing is NEUROCLAVE, developed at Lehigh University by Albin and Coulter (17) and trained by a model by Telikcherla (18). NEUROCLAVE was designed to operate both forward and inverse modes. The "forward" mode uses the neural network as a process model in an iterative process to determine the optimal autoclave setting for the next time step. The iterative process begins when a possible control action is sent to the neural network which predicts the final process parameters such as degree of cure and viscosity. These predicted parameters are compared to the desired output and an error is assigned to that possible path. Given the current time step is not complete, another possible control path is sent to the neural network and again a prediction is made. This continues until the end of the time step at which point the path with the lowest error is used as the control action in the next time step. One major drawback of the forward mode is in the error calculation which requires the relative importance of one process parameter over another throughout the cure, thereby making the optimization case specific (17).

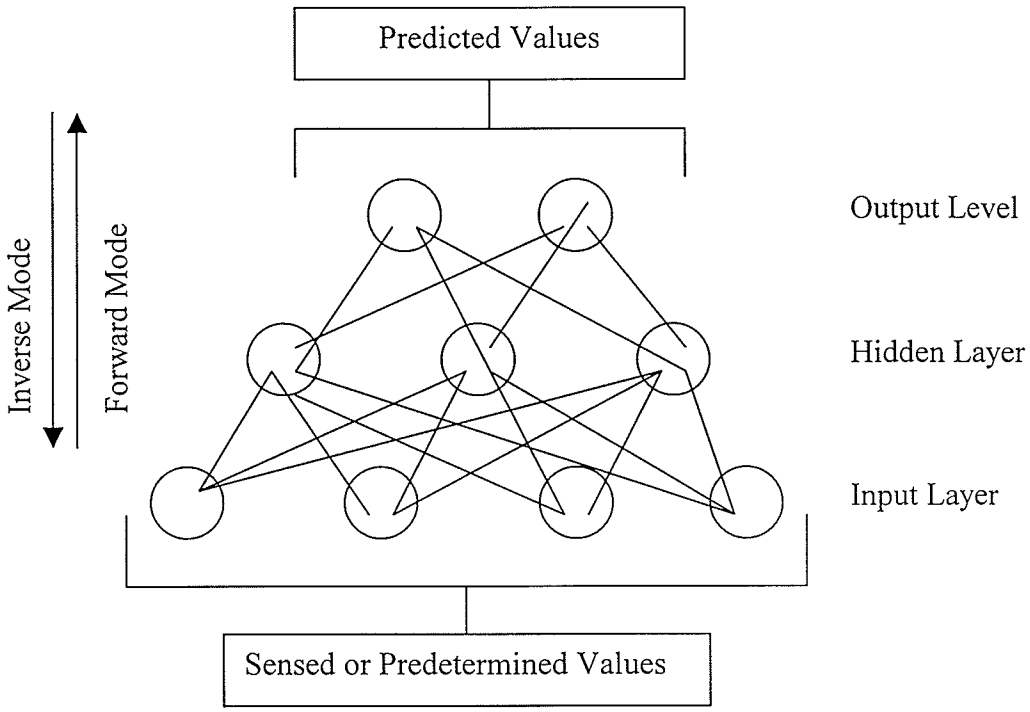

Figure 2.1 : Diagram of Simple Neural Network Structure

The inverse control mode is effectively a model that runs backward (17). It works faster than the "forward" mode as there is no iterative process. The controller works by taking the desired result and the current state of the cure to predict what the next action should be.

Albin and Coulter (17) found that the inverse mode worked the best. The forward mode was reported to be too heavily dependant on the weights assigned to various parameters in the error function. This makes the "forward" mode similar to an expert system in which the optimization is only as good as the information provided by the expert.

Neural networks do not appear to be flexible and reliable enough to meet the requirements of APCS. A major disadvantage to the neural networks is that accuracy is highly dependent on the training routine. Such training routines can be very time consuming and material dependant. Considering these factors, neural networks do not satisfy the requirements of APCS.

2.5 Heuristic Approach

Heuristic systems use a path generating program in order to assemble a large number of reasonable paths. Each of these paths are then evaluated with a heuristic/objective equation in an attempt to find the optimum within the set of paths.

Typically, the optimal cycle can be described as the one that results in the highest value for the objective equation.

Heuristics can be time prohibitive depending on complexity of the process model that the heuristic optimization program is linked with. It also depends on the number of paths that are to be evaluated. This can be demonstrated using an example. Assume that a cure cycle is 300 minutes long and is broken into four sections that each have certain requirements:

1. Ramp up – heat part to hold temperature isothermally (3 branches/interval, 100 minutes, 10 intervals)
2. Exotherm – minimize exotherm within part (5 branches/interval, 30 minutes, 7 intervals)
3. Hold – ensure the part is fully cured (3 branches/interval, 100 minutes, 5 intervals)
4. Ramp down – cool part to room temperature (2 branches/interval, 80 minutes, 5 intervals)

A branch would be a reasonable ramp rate such as 1 °C/min, 2 °C/min etc. Depending on the section of the cycle a branch could be positive or negative. Dividing the ramp up rate in 1 degree increments for analysis would generate 137,449 possible paths for evaluation. If the process model takes one minute to run a cure cycle, a solution will require approximately 2300 hours of computational time. This example is slightly

exaggerated as the number of iterations could be shortened by having less intervals or by creatively guiding the optimizer.

Once a path generation method is complete, an objective equation must be developed in order to evaluate each possible cycle. Pillai, et al. (19) have developed a heuristic program for cure cycle optimization. The objective equation is setup to assign higher values to cycles that produce parts with desirable properties. The objective equation can contain only parameters that are provided by the process model. Pillai, et al. describe the general form of a heuristic equation as (19):

$$h = f_1(A) + f_2(B) + \ldots + f_n(N) \tag{2.9}$$

Where:

A, B, N are all properties

Typically, the functions are written as power laws:

$$f_1 = p_1 * (A)^a \tag{2.10}$$

$$f_2 = p_2 * (B)^b \tag{2.11}$$

$$f_n = p_n * (N)^n \tag{2.12}$$

The function p and exponents a, b, and n in equations 2.10 to 2.12 are user defined to deliver the desired properties. This process to determine the optimal configuration is an iterative process.

In order to reduce the number of possible paths, Pillai divided the cure cycle into two sections, each with their own objective function (19). In section one, the objective was to get an isothermal steady-state part in the minimum process time. The heuristic parameters used in the first section were:

- t = process time required
- T_{eqlb} = temperature at which the part reaches the isothermal state, i.e. $T_{surface} = T_{center}$.
- T_{rate} = rate of temperature increase just prior to the part reaching an isothermal state.

Based on these parameters the general heuristic equation was defined as:

$$h = f_1(t) + f_2(T_{eqlb}) + f_3(T_{rate}) \quad (2.13)$$

where:

$$f_1 = p_1*(t)^a \quad (2.14)$$

$$f_2 = p_2*(T_{eqlb})^b \quad (2.15)$$

$$f_3 = p_3*(T_{rate})^c \quad (2.16)$$

It is important to note that the p functions and exponents a, b, and c in equations 2.14 to 2.16 were adjusted to obtain optimal cycle but were not given in the report (19). The first section of the cure was complete when the set dwell temperature was reached.

In section two, the objective was to complete a uniform cure with the smallest possible residual stress. The parameters used in the evaluation were:

- t_{exo} = the time at which the exotherm occurs.
- ΔT = the difference between the maximum temperature in part and the maximum autoclave temperature.
- δT = the difference between the surface temperature and the centerline temperature of the part.
- H ($\alpha_c - 0.5$) where α_c is the point at which cure at the cure at the center surpasses the cure at the surface. The intent of this function is to ensure that inside-out cure occurs and to indicate when the internal temperature is rising due to the exothermic curing reaction.

Therefore the general heuristic equation was:

$$h = f_1(t_{exo}) + f_2(\Delta T) + f_3(\delta T) + f_4(\alpha_c) \quad (2.17)$$

where:

$$f_1 = p_1*(t_{exo})^a \quad (2.18)$$

$$f_2 = p_2*(\Delta T)^b \quad (2.19)$$

$$f_3 = p_3*(\delta T)^c \quad (2.20)$$

$$f_4 = H\ (\alpha_c - 0.5) \quad (2.21)$$

It is important to note that the p functions and exponents , a, b, and c found in equations 2.18 to 2.20 were adjusted to obtain an optimal cycle but were not given in the report (19). It is also important to note that the f_4 function, in equation 2.21 is different than the other components. Pillai, et al. (19) set f_4 to be a step function to ensure inside-

out cure. This heavily penalized cycles that do not meet this requirement. This portion of the curing process was deemed to complete when the part was fully cured.

Heuristics are attractive in optimization because of the flexibility of the objective equation as well as the simplicity of setting one up. Results presented by Pillai, et al. (19) show that the optimizer reduced the cure from 300 minutes to 170 minutes. As well, the exotherm and residual stresses were shown to have decreased as well. However, the brute force method of searching through such a large group of cure cycles is time consuming and inefficient.

2.6 Genetic Algorithm

A genetic algorithm (GA) uses an objective equation in a manner similar to heuristics but applies a more intelligent searching method. Genetic algorithms work on the premise of "the survival of the fittest". As can be seen in Figure 2.2, a GA generates an initial, random population of cure cycles (first generation) and evaluates the fitness of each member of the population, one cycle at a time using the objective equation. Carrying over the top two fittest member of the previous generation(s), a new population (generation) is created. The first member of the new generation is the fittest member from the previous generation(s), the remaining members in the population are created by crossing over and mutating the top two members that were carried over from the previous generation(s). When a cross-over occurs, segments of one cure cycle are switched with segments from the other cycle. When a mutation occurs, the x and y coordinates of the

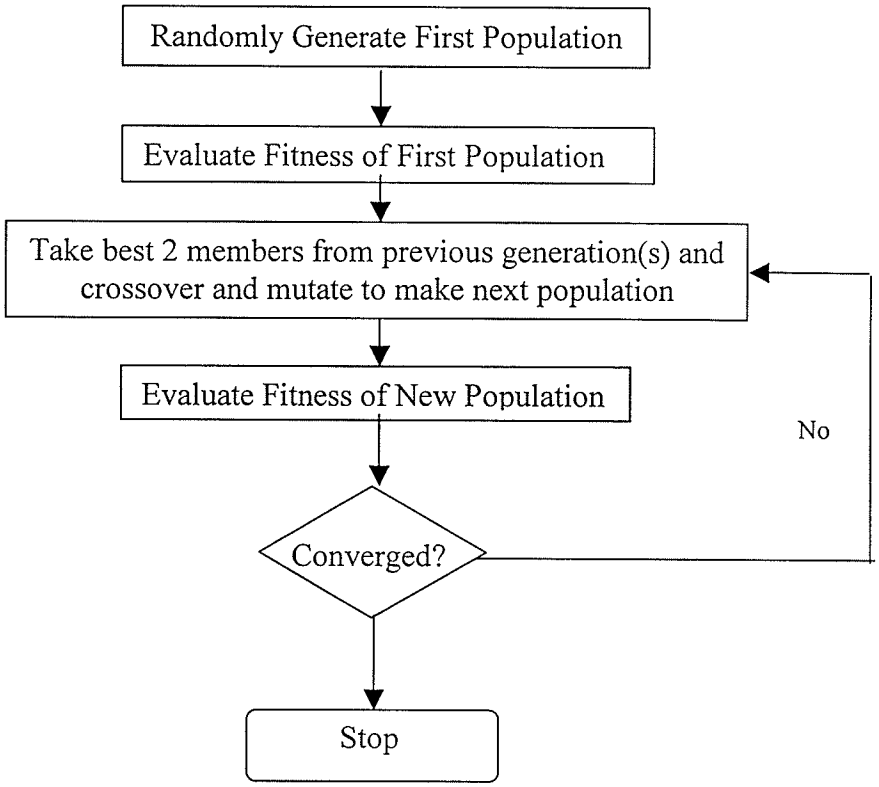

Figure 2.2: Flow Chart of a Genetic Algorithm

set points of a cure cycle (the time and temperature points) are randomly moved.

At each carry over, the best members of each population are compared to see if a convergence criteria is met. In general, a convergence criteria is either a set number of populations (generations) or when the improvement of the average fitness from one population to another is less than a set limit. If the convergence criteria is not met then the cycle continues until the criteria is met. Initial population sizes, objective equation, and convergence criteria must be carefully selected in order to ensure the proper operation of a genetic algorithm.

A modified version of a GA is a micro-GA. The micro-GA optimizes small groups of populations and then brings the best of these groups together to form the next set of populations. Tse and Chan (20), while optimizing airfoils, had shown that optimization times were 33% faster to achieve the same fitness using a micro-GA versus a regular GA. The operation of micro-GAs are described in more detail in Chapter 3.

Genetic algorithms require a fitness equation to evaluate each cure cycles generated by it. Fitness equation is just another term used for "objective equation" and "heuristics equation". The fitness equation used for the APCS was setup to meet a number of requirements as discussed in the next section. The general form of the fitness equation is similar to the heuristic equation in equation 2.9 as discussed in Section 2.4.

2.7 Summary of Literature Review and Thesis Approach

Initial process control systems were expert systems due to the limited computation power available at that time. In addition, composite processing science and hence process models were not well-developed. Although typically efficient in terms of implementation and computational requirements, expert systems are only as good as the expert knowledge used to develop the rules. Expert systems can be well suited for specific cases, say for a given lay-up and material type, but lack the flexibility required for APCS.

To take a more scientific approach to optimization and due to the availability of increased knowledge in processing science and in computational power, many model systems have been developed during the past two decades. Model systems typically require a large data base of empirical information such as material properties and heat transfer characteristics of the processing environment. The complexity of process models are dictated by the available mathematical representations used to simulate the process and the assumptions made with respect to the geometry (i.e. one, two or three dimensional representation of the composite part). The speed of the model systems increased with decreasing model complexity while the accuracy of the model systems increased with increasing complexity. Hence, model systems have to find a balance between accuracy and speed. The inherent inflexibility of expert based system is deemed to be too restrictive to be put into practical use. Hence, the APCS project has chosen a model based approach by choosing to use a genetic algorithm (GA). Similar to other model based optimizations, the APCS has attempted to strike a balance between speed,

taking into account the current state of computer technology and processing science accuracy.

Although, genetic algorithms have never been used directly to optimize cure cycles, GAs have been shown to be practical for optimizing other batch processes such as those found in the chemical processing industry (21). These algorithms have also been used in composite processing for optimizing gate and vent locations for resin transfer molding (22). Similar type optimizations have been performed for airfoil optimization (20). Although airfoils are quite different from cure cycles, both problems are represented by line segments. In the case of the airfoil, a closed section of line segments are used to define an airfoil shape and in the case of a cure cycle, the cycle is represented by a number of line segments that define a time-temperature curve. The genetic algorithm in both cases simply moves the connecting points around. In the case of airfoil optimizations, the points represent x and y coordinates while in cure cycles, points represent time and temperature. The GA moves the points until an optimal solution is found.

Genetic algorithms rely less upon expert knowledge and therefore are less case specific. Genetic algorithms also tend to solve faster versus a heuristic approach as the search path is more intelligent. Work on airfoil optimization using GA required 1×10^4 iterations, a factor of 10 less than the sample heuristic approach outlined above (20). Non-linear programming techniques appear to be case specific and unreliable in terms of convergence as compared to GAs.

In order to find an optimal cycle, a GA requires a fitness equation. The fitness equation used in APCS has been set up based on two criterion. The first criterion is that the fitness equation should incorporate current industrial quality measures. The second criterion is that the fitness equation should reflect current limitations of process model and sensors (i.e. the fitness equation should not require information that can not be supplied due to model and sensor limitations). The form of the objective equation will be similar to that used for heuristics as seen in equation 2.9.

Table 2.1 shows typical measures of quality found in the aerospace industry, the method by which the quality indicators are measured and the pass/fail criteria for each measure. The table also highlights equivalent measures for APCS system, virtual measurements and a possible fitness criteria for each measure.

Based on this table, the quality measures for the objective equation were chosen based on limitations of sensors and the process model. Several quality measures cannot be directly measured or modeled and so cannot be included in the objective equation until suitable sensors or models are developed. The objective equation and each component's shape are discussed in detail in chapter three.

The primary objectives of this thesis can be itemized as:

a) develop the optimizer module for the APCS

b) Integrate the optimization module with the other modules of APCS

c) Demonstrate successfully the optimization capability of the optimizer and APCS

A secondary objective is:

d) Material characterization for Hexcel F155 resin reinforced with Toho T300 carbon fiber for its physical, thermal, rheological and mechanical properties.

Industry Measure	Method of Measurement	Pass / Fail Criteria	APCS Equivalent Measure	APCS "Virtual Measurement"	Fitness Criteria
Material Properties	Indirect – Cure cycle conforms to a specified cure cycle*.	Cure cycle conforms to pre-specified standard.	Resin degree of cure	Direct prediction of resin degree of cure, α, at all composite material nodes.	Predicted α must reach a minimum specified value throughout the part.
Resin degradation (typically not a common problem in aerospace)	Visual / C-scan. Also, indirectly through measured part / tool temperatures.	Excessive measured temperatures. Appearance of any resin damage	Resin temperature	Either via maximum temp or an as-yet-undeveloped model. Based degradation prediction.	Maximum temp exceeded or "excessive" degradation as predicted.
Poor fit-up (part out of geometrical tolerance)**	Various, especially directly during assembly and via dimensional measurement.	Various – from "can we make it fit together?" to measurement of dimensions conformance to specifications.	Predict final part dimensions and, with a some modifications to COMPRO, fit-up loads.	Virtually "measure" distance between a set of points on a part or calculate fit-up loads or use an "excessive residual stress measure".	Measured dimensions or fit-up loads out of spec or excessive residual stress.
Imperfect compaction / resin flow.	Visual -observe areas with excessive resin or dry patches. C-scan	Various – typically based on relative ultrasonic signal attenuation	Resin content or thickness	Direct prediction of resin content.	Predicted resin content not within specification.
Excessive delamination	C-scan	Various – typically based on ultrasonic signal attenuation	None at present		

Fiber bridging	Inspection	Various – typically based on relative ultrasonic signal attenuation	None at present.		
Excessive void content	C-scan	Based on relative ultrasonic signal attenuation	None at present.		
Core crush	Visual only	Various from "Not acceptable" to "areas of specified maximum size allowed"	None at present.		
Marcelling (fiber waviness)	Visual only	Various from "Not acceptable" to "areas of specified maximum size allowed"	None at present.		

* - This process cycle is developed to ensure that adequate levels of resin cure are obtained throughout the structure with an "acceptable" margin of safety. One common way of reducing the length of the cycle that would be required to provide this assurance is to employ a lead / lag control system. This ensures via direct measurement that all areas of a component are subjected to an adequate time / temperature cycle.
** - The meaning of this and how it is measured will vary between organizations and with component type. For some components the standard may be as simple as "can components be fitted together using a load of X lbs force". For others, tolerances may be defined between given "hard points".

Table 2.1 : Relationship Between Aerospace Industry Standard Composite Quality Measures and APCS

CHAPTER 3

Optimization Program

3.0 Introduction

This chapter details the various components in the optimizer including the genetic algorithm, objective function and other interfacing modules such as the process model, SIMCLAVE and ARS (Autoclave Response Simulator). Following a description of the components, required inputs to the components and the component outputs are discussed. Finally, the overall function of the APCS is described.

An optimization problem has three essential components – design variables, objectives and constraints. The design variables are those for which the optimal values are sought. Since the program optimizes cure cycles, the design variables in this case are time and temperature. The objectives of the optimizer is to cure the part to a minimum degree of cure without causing the resin to degrade in the shortest possible time. The parameters that make of the objective function include time, temperature and degree of cure. Finally, the constraints used by the optimizer are a maximum time restriction and a maximum temperature restriction.

3.1 Structure of the Optimizer

As seen in Figure 3.1, the optimizer consists of two main modules, the path generator (genetic algorithm) and the Objective Function Module (OFM). In order for the optimizer to run, it requires several other interfacing modules. These include the process model, SIMCLAVE, and ARS (Autoclave Response Simulator). This section discusses each of the components and the manner in which they were assembled to make a working optimization program.

3.1.1 Genetic Algorithm

The genetic algorithm can operate in one of two modes. The first being the regular GA mode as discussed in Section 2.5. The second being the micro-GA mode as shown in Figure 3.2. The premise behind the micro-GA is that random cycles are injected continuously through each micro-cycle. This is unlike a regular GA where the only random population members are generated during the initial population (generation).

The micro-GA begins just as a regular GA does by generating a random population of cycles which are evaluated, one at a time as they are made, by the fitness function. For the remainder of this micro-cycle, the GA behaves as a regular GA. Once the micro-cycle is complete, the best member from the previous micro-cycle is carried forward to the first population of the next cycle. The remainder of this first population is then generated randomly and this micro-cycle again proceeds like a regular GA. These micro-cycles are repeated until the optimal cure cycle is found.

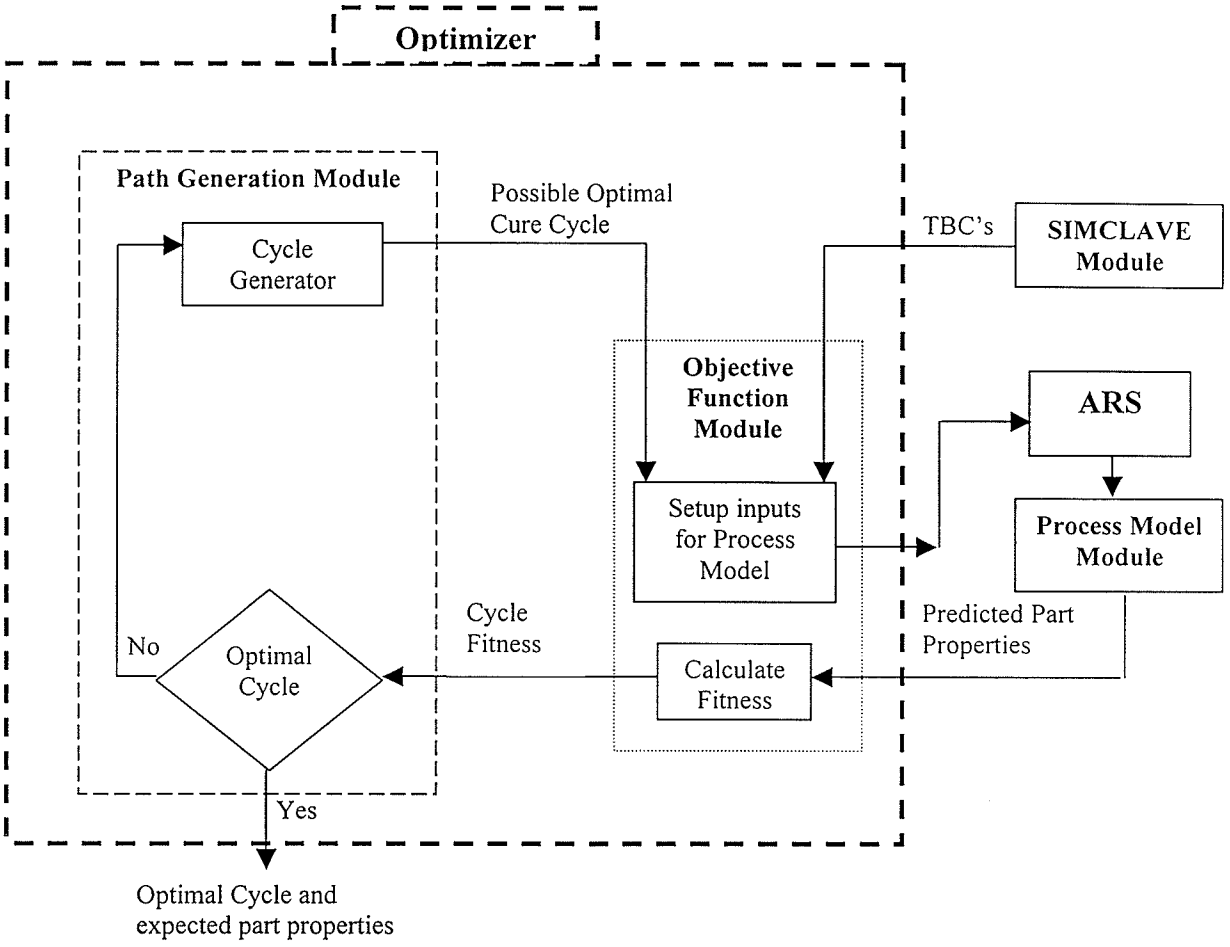

Figure 3.1: Functional Flow Chart of Off-Line Optimization Program

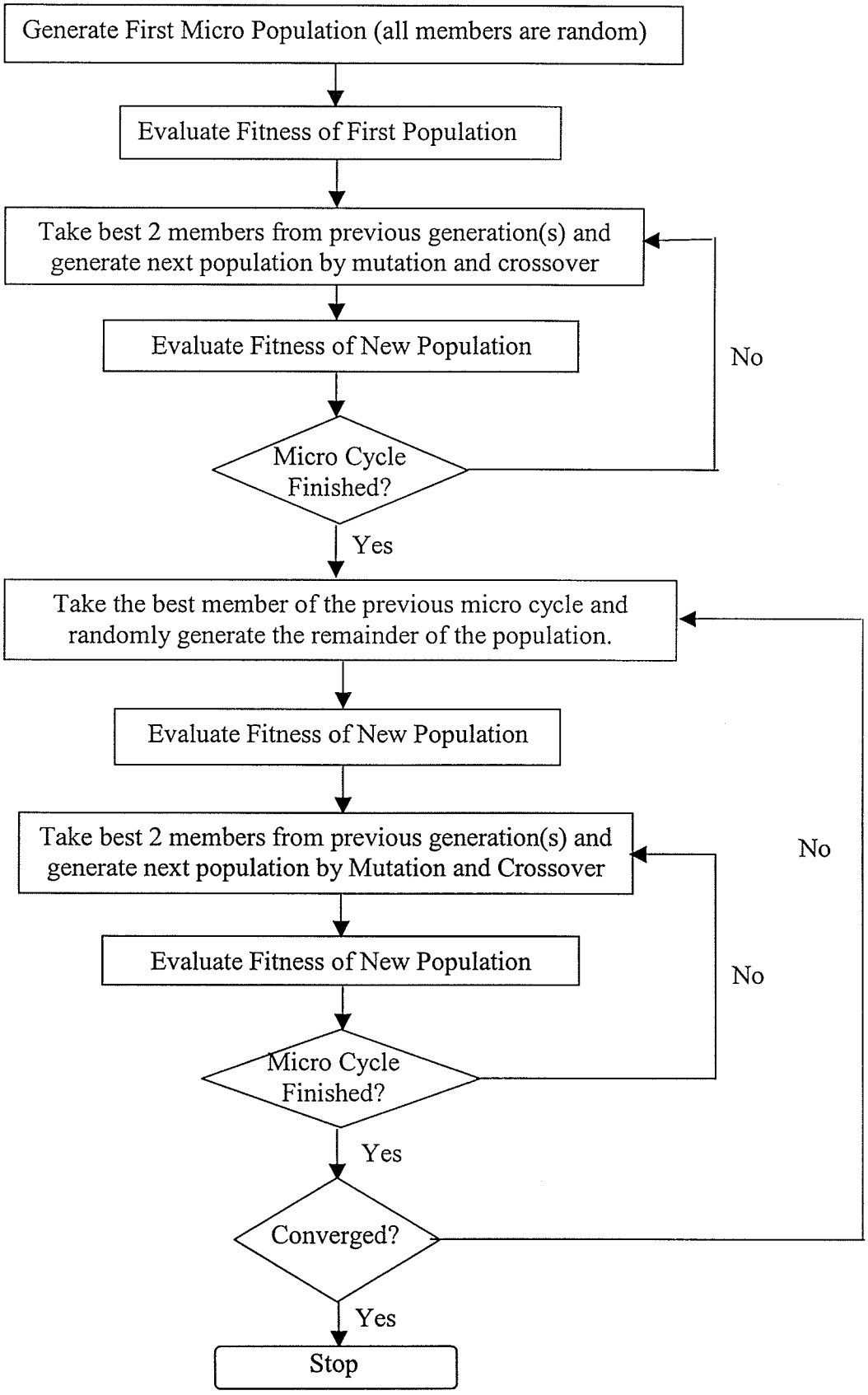

Figure 3.2: Flow Chart of a Micro-GA

Convergence for a micro-GA is similar to that of a GA in that convergence occurs at a set number of iterations or when the average fitness of two successive cycles is below a set value. In all cases for this thesis, a set number of iterations was used to determine the end point of the optimization. This was used for simplicity and to prevent the GA from getting stuck in a local minimum.

3.1.2 Objective Function Module

As mentioned previously, the fitness equation is required to evaluate the quality of a composite part processed using a given cure cycle. Hence, its form should reflect measures of quality used by the aerospace industry. Based on Table 2.1, quality parameters chosen for the equation are:

- Final degree of cure (α_{final})
- maximum temperature (T_{max})
- Cycle Time (t)
- Warpage (w)
- Temperature gradient (ΔT)
- Degree of cure gradient ($\Delta\alpha$)

The fitness equation involving all these quality factors is:

$$fitness = f(\alpha_{final}) + f(T) + f(warpage) + f(t) + f(\Delta T) + f(\Delta\alpha) \qquad (3.1)$$

where f is the function of a given parameter.

The shape of each function will be described in the following sections. These components were chosen primarily based on sensor and process model limitations and are similar to objective functions used by other researchers as discussed in Chapter2. The APCS program has the flexibility to switch on or off, one or more of these functions, depending on the problem that is being optimized.

All of the functions are designed to have a maximum value of 1 (FitnessPerfect) and a minimum of 0. Each of the functions are broken up into an "acceptable" region and an "unacceptable" region. A parameter is within the acceptable region when it meets or exceeds minimum requirements and is then assigned a fitness between FitnessAcceptable and FitnessPerfect. Parameters in the unacceptable region are assigned a fitness between 0 and FitnessAcceptable. Both FitnessPerfect and FitnessAcceptable are user definable parameters and are discussed in more detail in the next sections.

The shape of a function in the unacceptable region is typically exponential in order to drive the optimizer towards the acceptable region. Fitness functions in the acceptable region tend to be linear as there is no need to rapidly drive the solution towards the optimal condition as that may adversely effect the optimization of other parameters.

3.1.2.1 Shape of Degree of Cure Function – $f(\alpha_{final})$

In industry, it is imperative that a composite part be processed to a specified cure cycle. This ensures that all parts in the batch load have been fully cured regardless of their size and location in the autoclave which guarantees certain mechanical properties. The objective function accomplishes this by taking into account the minimum degree of cure within the part after a cure cycle is run. The shape of this function is shown schematically in Figure 3.3.

Mathematical form of this shape function is given in the following equations.

For $\alpha_{min} < \alpha < 1$

$$f(\alpha_{final}) = a\ \alpha_{final} - b \qquad (3.2)$$

where a and b are constants. The bounds for this relation are:

$f(\alpha_{final})$ = FitnessPerfect for $\alpha = 1$

$f(\alpha_{final})$ = FitnessAcceptable for $\alpha = \alpha_{min}$

For $\alpha < \alpha_{min}$

$$f(\alpha_{final}) = c \exp(d\ \alpha_{final}) \qquad (3.3)$$

where c and d are constants. The bounds for this equation are:

$f(\alpha_{final})$ = FitnessAcceptable for $\alpha = \alpha_{min}$

$f(\alpha_{final}) = 0.0001$ for $\alpha = 0$

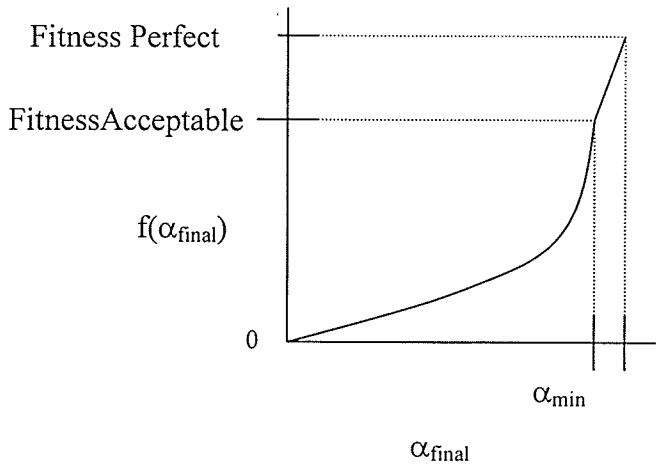

Figure 3.3: Shape of $f(\alpha_{final})$ versus Final Degree of Cure

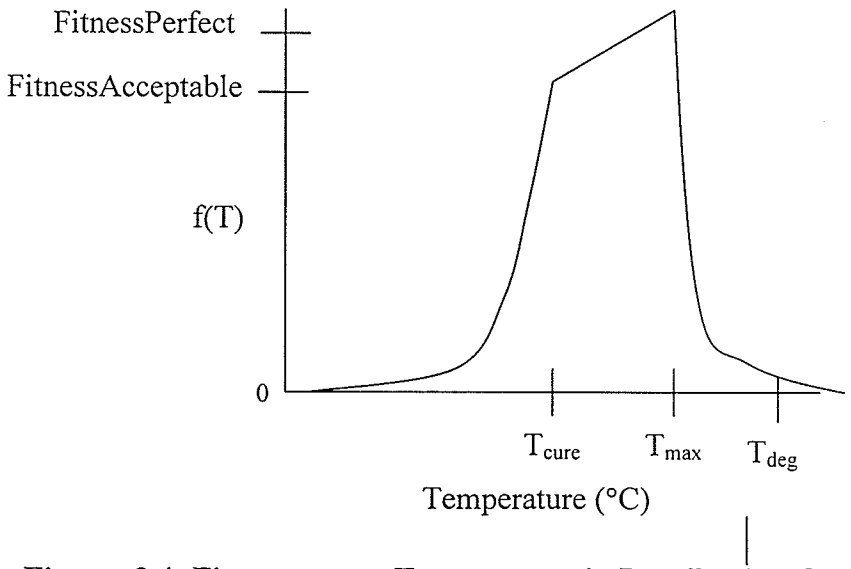

Figure: 3.4: Fitness versus Temperature in Part During Cure

With these assigned bounds, the values of constants a, b, c, and d are:

$a = (\text{FitnessPerfect} - \text{FitnessAcceptable}) / (1 - \alpha_{min})$

$b = a - \text{FitnessPerfect}$

$c = 0.0001$

$d = (1 / \alpha_{min}) * \ln (\text{FitnessAcceptable} / c)$

Minimum degree of cure (α_{min}), FitnessAcceptable and FitnessPerfect are user inputs. FitnessAcceptable and FitnessPerfect are the same for all fitness functions. This is so that the acceptable regions for all the functions have the same range of possible values. This prevents biasing of one function over the other. Typically, FitnessPerfect is set to 1 and FitnessAcceptable is set between 0 and 1.

At this time, the minimum degree of cure is taken to be the minimum degree of cure when the manufacturer's recommended cure cycle is run through the process model. The primary goal of this quality function is to ensure that the optimal cycle results in a minimum degree of cure in composite parts.

3.1.2.2 Shape of Maximum Temperature Function – f(T)

In a manner similar to degree of cure, maximum temperature can be used to represent resin degradation. A maximum temperature function is important because shorter cycles will tend to have higher cure temperatures and it must be ensured that the temperature is not too high. By including this function, the optimizer will tend towards an optimal cycle that will not cause the resin to degrade while ensuring a high cure

temperature and short cycle time. The shape of this function is shown schematically in Figure 3.4.

Mathematical form of this shape function is given in the following equations.

For $T_{cure} \leq T < T_{max}$ the objective function will be linear and assigned bounds of:

$f(T) = \text{FitnessPerfect for } T = T_{max}$

$f(T) = \text{FitnessAcceptable for } T = T_{cure}$

The form of the equation will be:

$$f(T) = aT - b \tag{3.4}$$

where a and b are constants

For $T < T_{cure}$ then the equation will be exponential and assigned bounds of:

$f(T) = \text{FitnessAcceptable for } T = T_{cure}$

$f(T) = 0.0001 \text{ for } T = \text{room temperature}$

The form of the equation will be:

$$f(T) = c \exp(dT) \tag{3.5}$$

where c and d are constants

For $T \geq T_{max}$ then the equation will be exponential and assigned bounds of:

$f(T_{max})$ = FitnessPerfect for $T = T_{max}$

$f(T_{max}) = 0.0001$ for $T >> T_{max}$

The form of the equation will be:

$$f(\alpha) = e \exp (f T) \tag{3.6}$$

where e and f are constants

With these assigned bounds, the values of constants a through f are:

a = (FitnessPerfect – FitnessAcceptable) / $(T_{max} - T_{cure})$

$b = a^* T_{max}$ – FitnessPerfect

$d = [1 / (T_{room} - T_{cure})] * \ln (0.0001 / \text{FitnessAcceptable})$

c = FitnessAcceptable * $\exp (-1 * d * T_{cure})$

$f = [1 / (T_{deg} - T_{max})] * \ln (0.0001 / \text{FitnessPerfect})$

e = FitnessPerfect * $\exp (-1 * f * T_{max})$

The maximum allowable temperature (T_{max}) is a user definable parameter and should be set below the resin degradation temperature (T_{deg}). This degradation temperature can be determined through differential scanning calorimeter tests or thermogravimetric analysis. The primary purpose of this function of the fitness equation is to ensure that cure takes place between the manufactures recommended dwell temperature (T_{cure}) cure and maximum temperatures (T_{max}). This will ensure a quick cure without causing resin degradation. The function is chosen as exponential in the unacceptable regions ($T < T_{cure}$ and $T > T_{max}$) so that the optimizer drives towards the acceptable region. The function in the acceptable region (between T_{cure} and T_{max}) is linear so that higher temperatures are not too heavily biased as to effect the optimization of other parameters once they are in the acceptable region.

3.1.2.3 Shape of Warpage Function - f(warpage)

Warpage can be defined as the change in dimension between two key points on the finite element model of the part. This is shown in Figure 3.5 for angle laminates, which is chosen as the case study for this thesis. This critical dimension L is an quality parameter. As a part warps, the distance between the nodes would increase or decrease. Other dimensional changes could include thickness changes.

Given coordinates (a_1, b_1) and (a_2, b_2) are the key points. The distance, x, between them is:

$$x = \text{sqrt}\ [\ (a_2 - a_1)^2 + (b_2 - b_1)^2\ ] \qquad (3.7)$$

The shape of the fitness function involving x is shown in Figure 3.6.

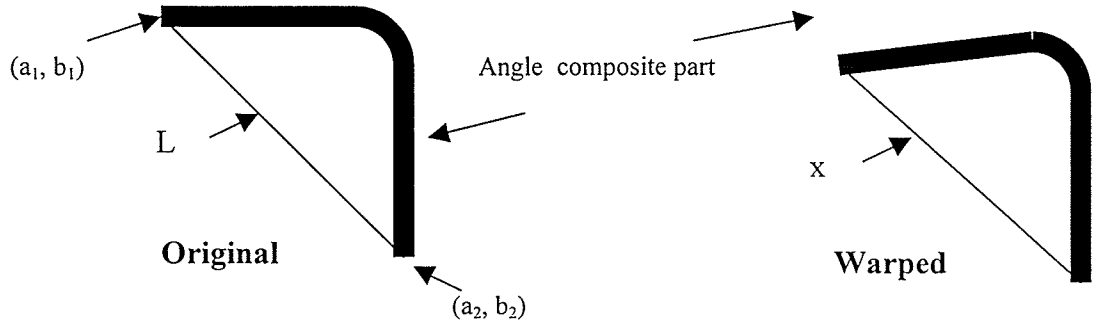

Figure 3.5 : Critical Distance "x" on Angle Composite Part

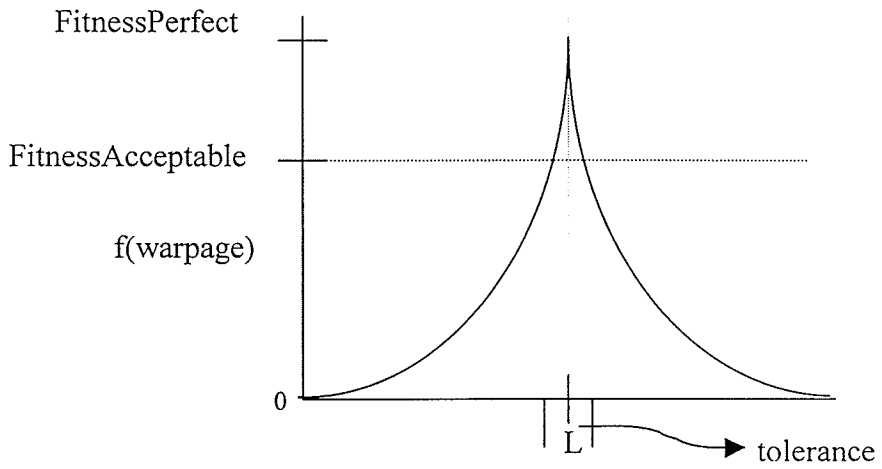

Figure 3.6 : General Shape of Objective Function versus Warpage (Distance Between Two Points)

This is mathematically represented as:

For x = L, the objective function will be linear and assigned bounds of:

f(w) = FitnessPerfect for x = L

f(w) = FitnessAcceptable for x = L ± tolerance

The form of the equation will be:

$$f(w) = a \exp (b\, x) \tag{3.8}$$

where a and b are constants

With these assigned bounds, the values of constants a and b are:

b = (1 / tol) * ln (FitnessAcceptable / FitnessPerfect)

a = FitnessPerfect * exp (-1*b*L)

The equation above is designed to have a symmetrical tolerance (tol) i.e. one that has the same value for the positive and negative tolerance. The equation could be designed such that the positive tolerance is different than the negative tolerance.

As another measure of quality, warpage has been included in the objective function. By including warpage, cure cycles with excessive warpage will be assigned

lower finesses so that the genetic algorithm will tend towards cycles that deliver lower warpage. Exponential functions are chosen as it helps to drive the optimizer towards the acceptable region. The exponential function is extended into the acceptable range as the tolerance is usually small and this part of the exponential function will act effectively as a linear functions.

Unfortunately, the warpage predictions of the process model used in this study were not satisfactory and so this component of the fitness function was turned off during optimization but is included for future versions of the APCS when a more suitable model for warpage prediction is found.

3.1.2.4 Shape of Time Function – f (t)

As a component of the objective function, time is included as it is extremely important to industry to reduce cycle time and therefore costs. The shape of this function is shown schematically in Figure 3.7.

Mathematical form of this shape function is given in the following equations.

For $t < t_{manufacturer}$, the function will be represented by a linear function and have the bounds:

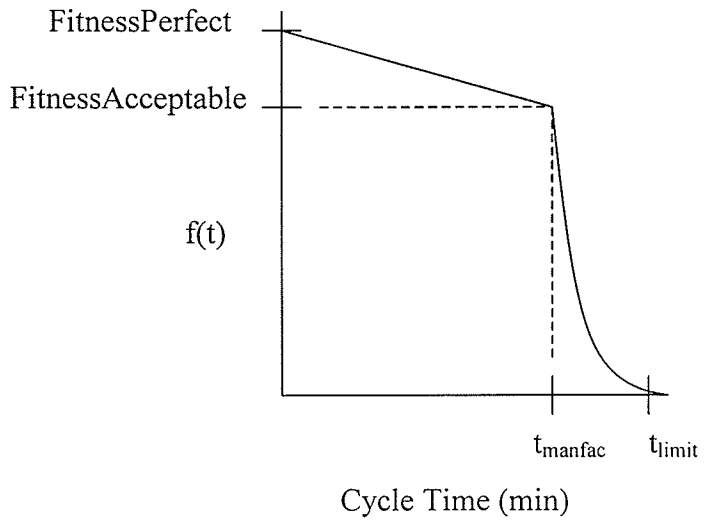

Figure 3.7: Shape of Time Fitness Function – f(t)

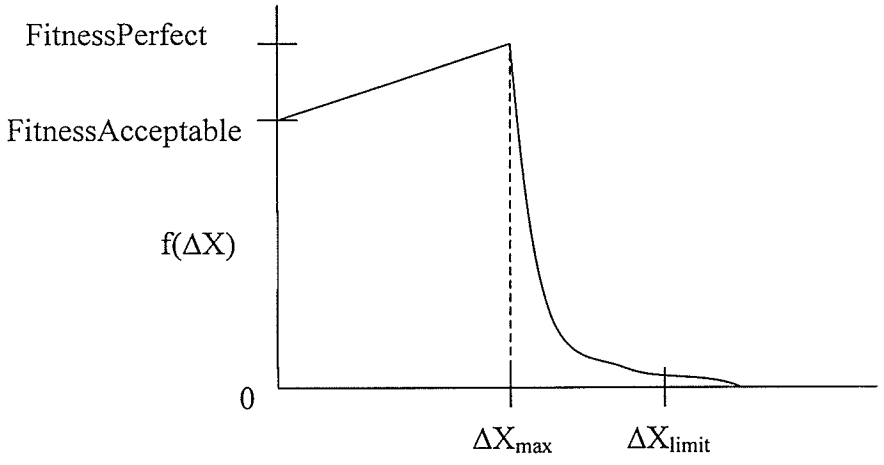

Figure 3.8 : Fitness Function for Temperature and Cure Gradient

(Note that "X" can be replaced with "T" or "α")

$f(t) = \text{FitnessPerfect for } t = 0$

$f(t) = \text{FitnessAcceptable for } t = t_{manufacturer}$

The form of the equation will be:

$$f(t) = at + b \tag{3.9}$$

where a and b are constants

For $t > t_{manufacturer}$, the function will be exponential and have the following bounds:

$f(t) = \text{FitnessAcceptable for } t = t_{manufacturer}$

$f(t) = 0.0001 \text{ for } t >> t_{manufacturer}$

The form of the equation will be:

$$f(t) = c \exp(d\, t) \tag{3.10}$$

where c and d are constants

With these assigned bounds, the values of constants a, b, c, and d are:

b = FitnessPerfect

$a = (FitnessAcceptable - b) / t_{Manufac}$

$d = [1/(t_{Limit} - t_{Manfac})] * \ln (0.0001 / FitnessAcceptable)$

$c = FitnessAcceptable * \exp (-1* d *t_{Manfac})$

The user is required to enter three times:

1. The manufacturer's recommended cycle time ($t_{manufacturer}$). This is the maximum reasonable time as a cycle that takes longer than this is no use to industry.
2. The limit time (t_{limit}). This is a time past the manufacturer's time so that equation 3.10 can be calculated. This is required to make the fitness function continuous over the range of all possible values
3. The final input is the minimum time (t_{min}), used for version 2 of the time function as described in chapter 4, which is the shortest possible time for complete curing given the part was instantaneously brought up to the maximum temperature (T_{max}). This is determined via the use of the cure kinetics model of the material.

The objective of this fitness function is to drive the GA towards shorter cure cycles.

3.1.2.5 Shape of Temperature and Cure Gradient Functions - f(ΔT), (Δα)

The purpose of these two components is to indirectly reduce the stress and warpage. From other studies, as discussed in chapter 2, it was seen that setting a maximum acceptable ΔT and maximum acceptable Δα was an effective way to control the exotherm and to reduce stress. In order to cure the part as fast as possible, the fitness function for this component is designed so that the maximum acceptable cure and temperature gradients deliver the highest fitness. If the equation was designed to minimize the gradients then the optimizer would tend towards a very long cycle, which is undesirable. Figure 3.8 depicts the shape for the temperature and degree of cure gradient component of the fitness function.

Mathematical form of this shape function is given in the following equations. Note that X represents both "T" or "α" as the shape of the functions are identical

For $\Delta X < \Delta X_{max}$ then the function will be linear and have the bounds:

$f(\Delta X)$ = FitnessAcceptable when $\Delta X = 0$

$f(\Delta X)$ = FitnessPerfect when $\Delta X = \Delta X_{max}$

The form of the equation will be:

$$f(\Delta X) = a*\Delta X + b \tag{3.11}$$

where a and b are constants

For $\Delta X > \Delta X_{max}$ then the function will be exponential and have the bounds:

$f(\Delta X) = 0.0001$ when $\Delta X = \Delta X_{limit}$

$f(\Delta X) = FitnessPerfect$ when $\Delta X = \Delta X_{max}$

The form of the equation will be:

$$f(\Delta X) = e \exp (f^*\Delta X) \tag{3.12}$$

where c and d are constants

With these assigned bounds, the values of constants a, b, c, and d are:

$b = FitnessAcceptable$

$a = (FitnessPerfect - FitnessAcceptable) / \Delta X_{max}$

$f = \ln(FitnessPerfect / 0.0001) / (\Delta X_{max} - \Delta X_{limit})$

$e = 0.0001 / (f * \Delta X_{limit})$

The user is required to input the maximum allowable gradient (ΔX_{max}) and a limit value (ΔX_{limit}) that is far from ΔX_{max} so that a continuous function is available to accommodate all possible values. This fitness function was not used for the thin angle laminate case that is tested for this thesis. As the part was thin enough, temperature and

cure gradients were small enough without having to take them into consideration. It has been programmed into the optimizer for future work with thick laminates.

3.1.3 Interfacing Modules

Several other modules are required for the optimizer to function. As seen in Figure 3.1, a process model, SIMCLAVE and ARS are all required to make the optimizer work well.

3.1.3.1 Process Model

As APCS is taking a model based approach to optimization, a reliable process model is crucial. The first process models to be developed were capable of only predicting one or two curing parameters, namely temperature and cure (23, 24). Typically these models were only one-dimensional due to computational restriction during the model development. As computer power began to grow, so did the complexity and dimensionality of the process models. Even with current desktop computer technology, most models remain two-dimensional but prove effective for a much broader scope of problem as compared to one dimensional models.

More sophisticated models use a sub-model approach such as those developed by Bogetti and Gillespie (25, 26), and White and Hahn (27, 28). Building upon previous models, The University of British Columbia developed COMPRO (7). COMPRO, uses a

two dimensional mesh of the composite part. COMPRO was the process model used for this thesis but in the future a University of Manitoba model will be used by the APCS. COMPRO has three main modules - thermochemical, flow, and stress. Each of the modules are briefly discussed in the next three sections but a more detailed description of each module please refer to Johnston (7).

1. Thermochemical Module

The thermochemical module is used to determine the temperature and degree of cure within the part. It takes into account the energy exchanged through convection between the autoclave and the part, as well as, energy generated from the kinetic reaction in the resin. This module is heavily dependant on the heat transfer in the autoclave and so characterization of the heat transfer within the autoclave is important. SIMCLAVE has been developed to predict heat transfer into the part. This program is discussed in more detail in Section 3.1.3.2.

2. Flow Module

The role of the flow module is to predict resin flow and compaction. The flow module requires temperature and cure data from the thermochemical module to calculate the viscosity of the resin which in turn is used to predict resin flow. It has been shown that both resin shrinkage and resin modulus development have an effect on residual stress in a part (27). Other problems can arise when there is too much or too little resin flow.

Variations in resin content can cause thickness differences at a various locations in a component which can lead to residual stress or poor fit with another part (7). Resin poor areas may not have desired mechanical properties as a composite requires both the fibers and matrix to function properly. Porosity can occur in resin poor areas which could be detrimental in a honeycomb structure in which fluids could be absorbed.

3. Stress Module

The stress module is used to determine residual forces and deformation in a part. Process models such as COMPRO are able to simulate part removal from a tool (7). This is important as residual stresses can build up in a part because of the mismatch in coefficient of thermal expansion (CTE) between the tool and part and the tool-part surface interaction. The tool-part interaction was initially simulated in COMPRO via a shear layer in this thesis. Such a technique has been tried in the past (7). However, this is not a reliable method and warpage predictions for initial runs of the angle laminate used in this thesis were poor. The understanding of tool-part interaction is not fully developed and research at various universities are working on this problem. It is hoped that a reliable warpage prediction model will be incorporated into APCS in the future. Until such time as a more suitable model is found, the stress module is turned off in the process model during optimization runs for this thesis. This also means that there was no need to run the flow module either as only cure and temperature were required for the final form of the fitness equation.

3.1.3.2 Heat Transfer Predictions – SIMCLAVE and ARS

As mentioned previously, heat transfer predictions are crucial to the effective operation of the process model. Accurate heat transfer predictions rely on two main factors – heat transfer coefficient and air temperature. SIMCLAVE is being developed to accurately predict the heat transfer within the autoclave, while another program ARS (Autoclave Response Simulator) is being developed to predict the actual air temperature given control commands sent to the autoclave.

In the past, some research into autoclave heat transfer predictions has been purely computational in nature such as studies by Telikicherla, et al. (29). The curing process was simulated assuming the autoclave was a straight channel with an obstruction and that the cure assembly was a near-flat plate geometry. The model required 8 hours of CPU time on a CRAY YMP computer. Such a method is too computationally intense for use with the APCS and is the main reason for the experimentation which lead to the empirical models used in SIMCLAVE.

Other researchers such as Ghariban, et al. studied airflow in a 10:1 scale model of an unpressurized autoclave (30). Unfortunately, the model was not pressurized and therefore did not fully simulate the actual autoclave environment. Pressure has been shown to be an important factor in heat transfer. In fact, it was found that heat transfer was dependant on pressure and an empirical constant, C, that was unique to each autoclave (7, 31). This relation is shown in equation 3.13.

$$h = C\left(\frac{P}{T}\right)^{\frac{4}{5}} \tag{3.13}$$

This model for the heat transfer coefficient was used in the process model for the optimization runs. Angle laminates that were built to validate the heat transfer predictions of the process model were cured in the same autoclave as was used for experiments to characterize the heat transfer in the autoclave (Hudek, 9). It was from Hudek's models that the coefficient "C" was taken from for the process model.

As mentioned earlier, the second important factor in predicting heat transfer into a part is the accurate prediction of the autoclaves response to control commands and therefore the accurate prediction of air temperature. At the beginning of a cure cycle, a control command such as "ramp at 3°C/min" cannot be followed exactly. It takes time for the heaters to warm up and it is only after a period of time that the autoclave can achieve a ramp rate of 3°C/min. This is depicted in Figure 1.3. Therefore, the ARS program is being written to take the "hard" control points generated by the optimizer and turn them in to the actual air temperature that a part will be subjected to. This program was not finished at the time that this thesis was completed and so air temperature is predicted by a crude model that is built into the process model.

3.2 Operation of the Optimizer

Now that each of the components have been described, this section details how all components were linked to optimize a cycle.

3.2.1 Program Flow

Optimization progresses through components as shown in Figure 3.9. The first step is to complete the pre-processing before the optimization routine is begun. The pre-processing steps are as follows:

1. Setup process model – finite element mesh, boundary conditions, material properties, etc.
2. Setup APCS inputs required for optimization via APCS interface (currently options are changed via a text file)
3. Run SIMCLAVE to set boundary conditions on process model
4. Start Optimizer

Once the optimizer is started, the following steps are taken:

1. Generate a cure cycle
2. Output cycle to a text file
3. Call objective function module
4. Run Autoclave Response Simulator (ARS)

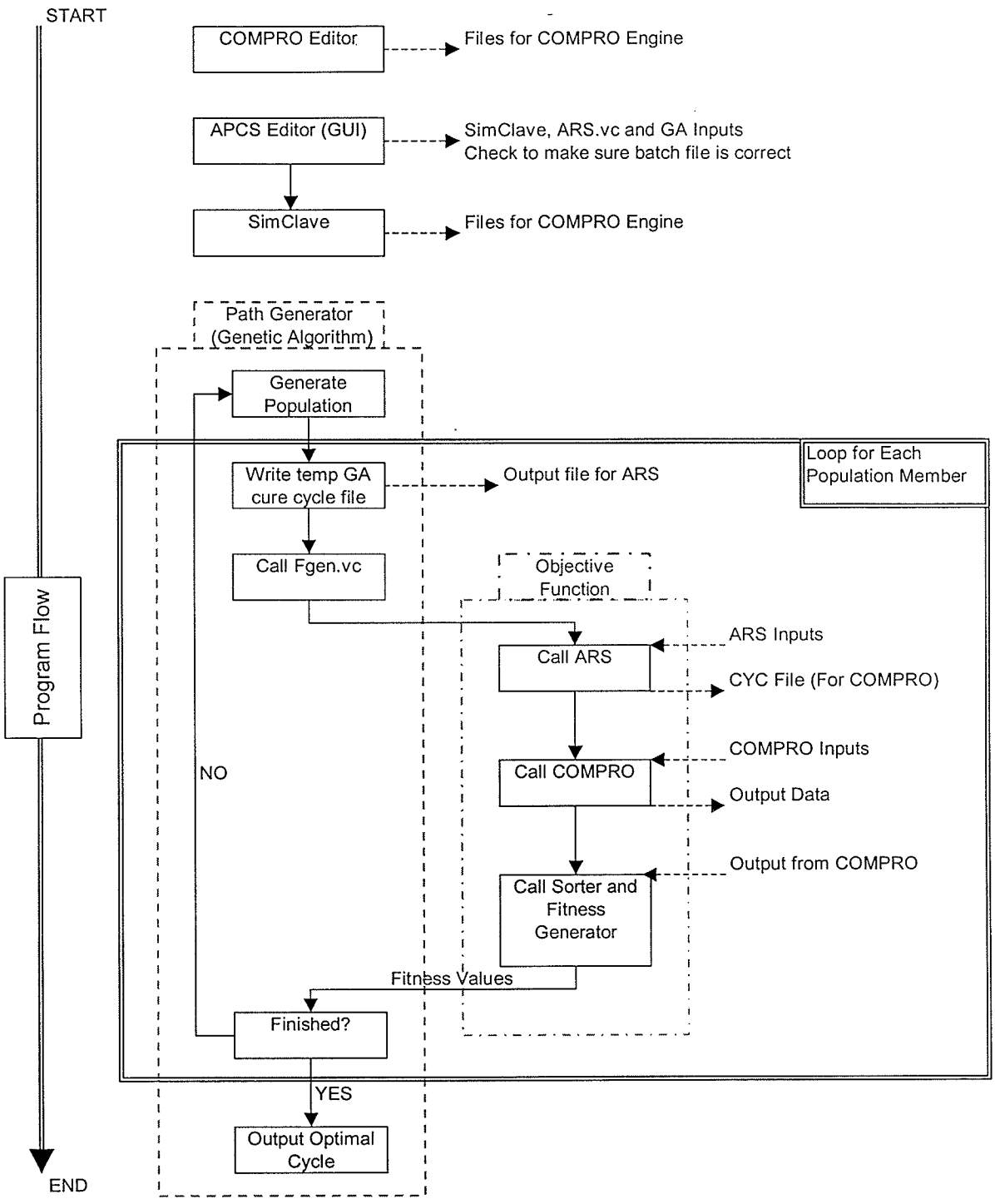

Figure 3.9: Schematic of Optimization Program Order of Operation

5. Run process model – (in the case of this thesis – COMPRO)
6. Run process model output sorter
7. Calculate fitness and write fitness value to a text file
8. Return to GA which reads in fitness value

How the program loops depends on the options set in the optimizer in terms of whether the optimizer is running in GA or micro-GA mode.

3.2.2 Program Data Flow

Please refer to Figure 3.10 for a schematic of the data flow. As mentioned previously, the first step in the pre-processor is to set-up the process model, in this case, COMPRO. The files required to be setup for COMPRO are:

- .MAT file (COMPRO material data file)
- .CTL file (COMPRO control file)
- .PRJ (COMPRO project file for a given model)
- .LAY (COMPRO material lay-up sequence file)
- .BCH (COMPRO batch file – i.e. the project to be run)
- .BCI (COMPRO boundary conditions file)
- .PAT (COMPRO finite element mesh file)

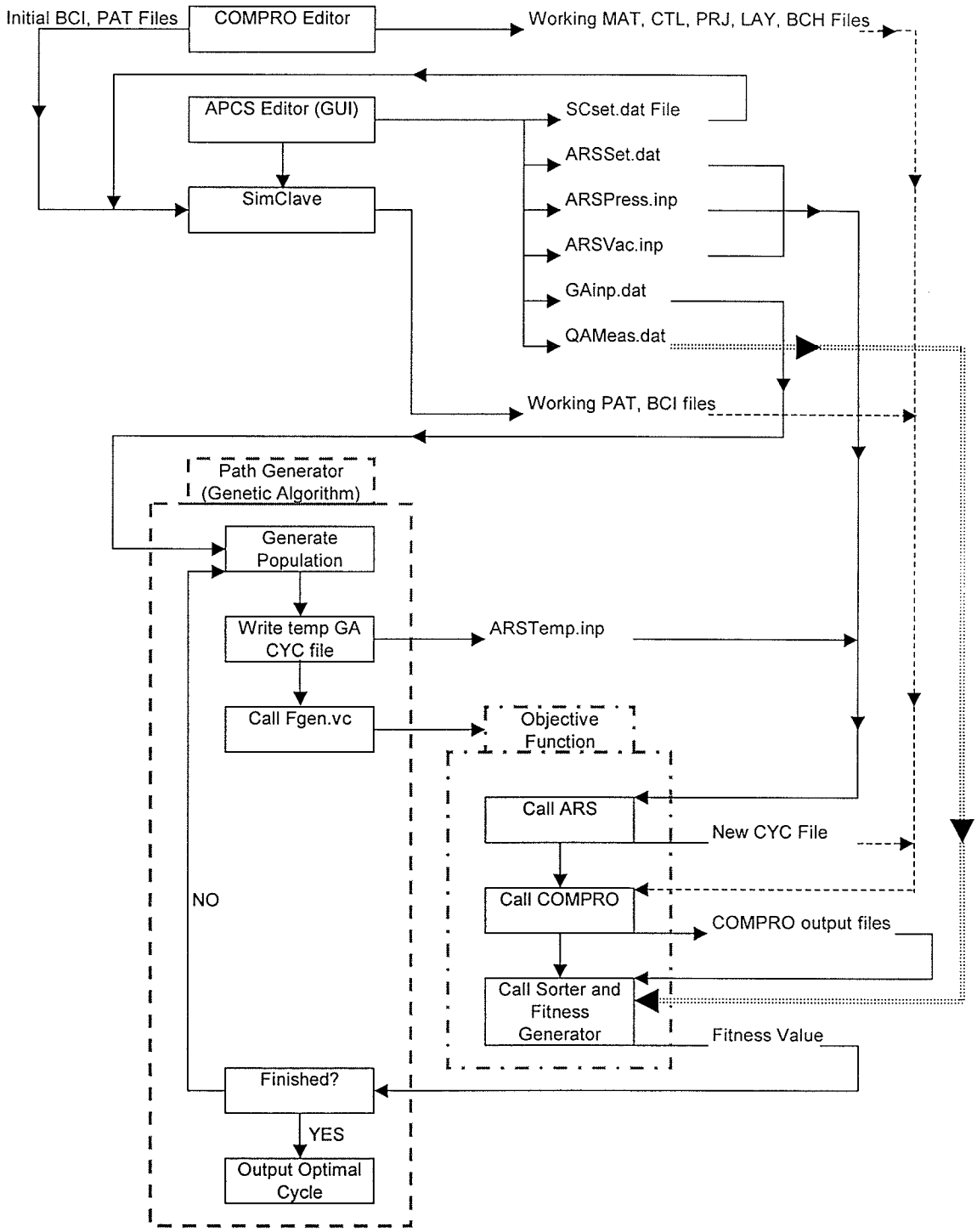

Figure 3.10: Schematic of Data Flow in Optimization Program

These files will be used without modification except the .BCI and .PAT files which are modified via SIMCLAVE to accurately represent the heat transfer boundaries within the autoclave.

Once the finite element model of the part is made, all the options for the optimization must be set in the following files:

- Scset.dat (sets the options required for SIMCLAVE)
- ARSset.dat (sets the options for ARS)
- ARSPress.inp (sets the pressure cycle for the cure cycles)
- ARSVac.inp (sets the vacuum cycle for the cure cycles)
- Gainp.dat (sets the options for the genetic algorithm)
- QAMeas.dat (sets the parameters for the fitness function, these are the user defined parameters as detailed in section 3.1.2.x)

Following the setup, SIMCLAVE is run in order to properly setup the .PAT files and .BCI files, which completes pre-processing. Once the optimizer is started, by starting GAMAIN.EXE, an initial cycle is generated which can be either random or a user defined cure cycle. This cycle is tested against a set of absolute constraints of time and maximum temperature as set in Gainp.dat. If a cycle exceeds the maximum time or a maximum temperature at any time, then there is no need to run the cycle through the process model and that cycle is assigned a low fitness value by the GA. Only after a cycle passes this test is it written out as ARSTemp.inp file. At this point the Objective

function Module (OFM) is called by the GA. The fitness function calls the ARS program to combine the ARSTemp.inp and the other ARS associated files into a .CYC file format for use by COMPRO. For more information on ARS, please refer to Michael Hudek's thesis (9). Once the new .CYC file is written, the COMPRO engine is started which runs COMPRO for the part model listed in the .BCH file.

Upon completion, the output sorter is started to sort through the COMPRO output files for the data that is required to calculate the fitness value for the cycle. Parameters extracted from the COMPRO outputs include degree of cure, temperature and nodal displacements. This data, as well as the information in QAMeas.dat, which contains the user defined parameters for the fitness functions, are used by the fitness functions to calculate the fitness value for the current cure cycle. A single fitness value is output as a text file and control returns to the GA which reads the fitness value from the text file. Depending on the stage of optimization, the GA either continues to generate the next cure cycle and continue the loop or it stops and writes out the optimal cure cycle to a text file. All optimization runs used in this thesis used a fixed number of iterations as the end point for the optimization.

3.2.3 Programming Notes

The programs listed above have been written by several programmers in several different computer languages. The following is a list of programs and the languages in which they were programmed.

COMPRO – Visual Fortran

APCS Editor – to be programmed in Visual Basic

SIMCLAVE – Visual C++

Genetic Algorithm – Visual Fortran

Objective Function – Visual C++

With all the different languages, it was decided that communication between programs would be via text files. All files mentioned previously are simple text files that are read and interpreted by each program..

CHAPTER 4

Experimental and Simulation Details

4.0 Introduction

This chapter describes the tests that were required to characterize a composite material. As well, details of process model and optimization settings are given.

4.1 Material

Material characterization was done for Hexcel F155 resin with Toho T300 carbon fiber. However, material data for this composite was not complete early enough for use with the process model. Instead, the process model used material data for Cytec-Fiberite 934 resin with Torray T300 fibers. Details on this material are available in the MSc. thesis of Koteshwara (32). Both materials are commonly used in aerospace industry.

4.2 Experimental Details

Material characterization was performed to determine the following material properties:

- Cure Kinetics parameters
- Resin Degradation Temperature (T_{deg})
- Cure Shrinkage
- Coefficient of Thermal Expansion (CTE)

- Specific Heat Capacity
- Viscosity
- Gel point

Each of the testing procedures are described in the following sections.

4.2.1 Cure Kinetics

Cure kinetics parameters are used by a process model to determine the degree of cure as well as the heat evolved from the exothermic reaction when the resin cures. A TA Instruments 2910 MDSC (Modulated Differential Scanning Calorimeter) was used to perform cure kinetics experiments. The heat flow measurements from the MDSC are used to determine the degree of cure (α) which is defined to be:

$$\alpha = \frac{H(r)}{H(tot)} \tag{4.1}$$

where:

H(r) = heat evolved over a given time period

H(tot) = Total possible heat that could be evolved

Samples were made from woven pre-impregnated composite (prepreg). Samples were removed from the freezer and warmed to room temperature in a vacuum chamber at which point they were cut into 10-15 mg samples. Specimens were then weighed and

crimped into a non-hermetic aluminum pan. At the same time, a reference aluminum pan without composite and of approximately the same weight as the sample pan was prepared.

Two types of "runs" were performed – isothermal and dynamic. For an isothermal experiment, the test cell was first preheated to the isothermal temperature. Once preheated, the test cell was opened and the reference and sample pans were placed inside the cell. Once the pans were in place, the cell was closed rapidly and the test was allowed to continue. For a dynamic scan experiment, the sample and reference pans were placed into the cell before heating or cooling began. Once the samples were inside, the cell would then be closed and the experiment would begin. Experiments on the samples fell under four possible test groups:

Test 1 (Test for Total Heat of Reaction):

Dynamic scans were performed on three samples at 2°C/min in order to determine the total heat of reaction. Information from this test was used to determine the total heat of reaction, H(tot), in equation 4.1. This information is required to determine the degree of cure in subsequent isothermal tests.

Test 2 (Isothermal Curing):

Initial isothermal runs with uncured samples were performed at 90, 100, 110, 120 and 130°C. Two samples were run at each temperature. Data from test 2 is required to

determine the degree of cure versus time for a given isothermal temperature. Samples run through this test were also run through tests 3 and 4.

Test 3 (Completely Cure Sample from Test 2):

After initial isothermal runs, the samples were subjected to a dynamic run at 5°C/min from room temperature to 200°C to ensure the sample had completed curing before moving on to the next test. The sample was finished curing when the heat flow curve returned to the baseline which indicates that the reaction was complete.

Test 4 (Re-run Sample from Test 3 Through Isothermal Cycle):

Fully cured samples from test 3 were re-run through their respective isothermal run as performed in test 2. It was found that when the test cell was pre-heated and then opened to place the samples inside, several minutes lapsed before the cell equilibrated. This shifted the baseline during testing which was compensated for by this test. .

Data from the MDSC was output as a curve of heat flow (W/g) versus time (min). The total heat evolved from test 1 was measured by integrating the area under the heat flow versus time curve. The result is the total heat evolved H(tot) in J/g (see Figure 4.1).

Data from tests 2 and 4 were used to calculated dα/dt as a function α. The model used to fit the data is:

$$\frac{d\alpha}{dt} = k * \alpha^m * \left(\alpha_{final} - \alpha\right)^n \qquad (4.2)$$

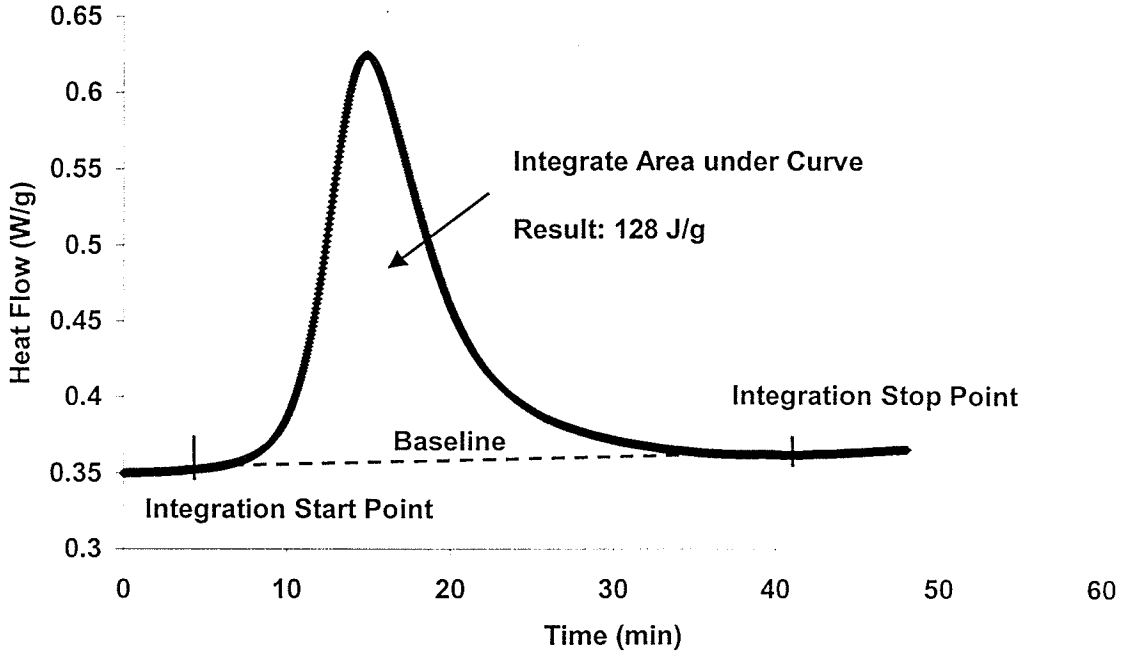

Figure 4.1: Total Heat of Reaction (Area Under Heat Flow vs. Time Plot)

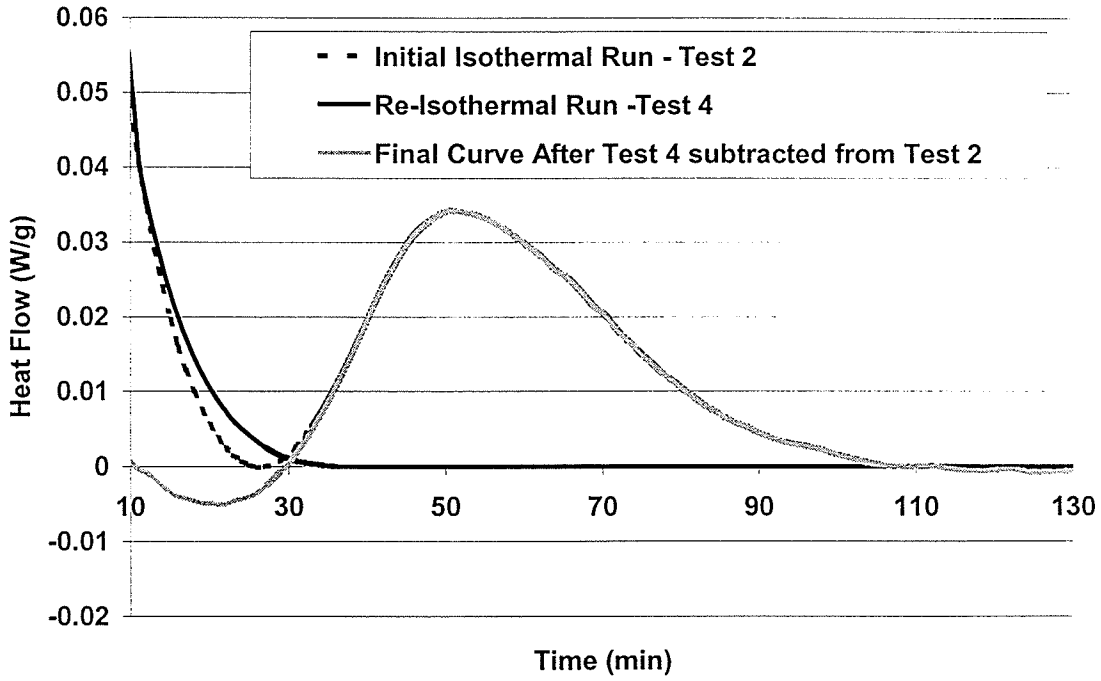

Figure 4.2: Plot of Initial Isothermal Run, Re-Isothermal Run and Final Curves

Where:

$k = Z \exp(-E/RT)$

Z is in sec^{-1}

E has units of kJ/mol

T is temperature in Kelvin

R is the universal gas constant

α = degree of cure

m is constant and dimensionless

$n = A \exp(B*T)$

A is dimensionless

B is in $Kelvin^{-1}$

T is temperature in Kelvin

$\alpha_{final} = C \exp(D*T)$

C is dimensionless

D is in $Kelvin^{-1}$

T is temperature in Kelvin

The steps to determine the coefficients for equation 4.2 are as follows:

1. The initial isothermal run file was open (from test 2) in TA Instruments Universal Analysis Software and was shifted such that the section where the baseline ends (cure is finished) was set to 0. The re-isothermal run (test 4) file was then opened and shifted such that the curve was 0 at the same point in time as the initial

isothermal run. The re-isothermal curve was used to compensate for the thermal lag in the equipment and was subtracted from the initial isothermal run. The new curve 's heat flow (mW), temperature (Celsius) and time (minutes) data was output in a tab delimited text file so it could be read by other programs. The time at which the reaction started and stopped was noted as it is required for step 2. Refer to Figure 4.2 to view all 3 curves.

2. The data set for the new curve was then opened in Excel so that:
   - Time could be changed from minutes to seconds
   - Data before and after the reaction was removed
   - The time was reset so that the beginning of the reaction started at time = 0.

3. The data was again saved as tab delimited and opened in Kaleidagraph software. Here the following was done:
   - Heat flow was converted from mW to W
   - A check was performed to make sure all heat flow values were positive, if not then the heat flow curve was shifted so that all values were positive (negative values mean negative energy flow which is not possible for an exothermic reaction)
   - Integrate heat flow (W) with time (sec) to get energy (J)
   - Divide "running" energy value at each time step by total energy possible (as found by test 1) in order to calculate degree of cure (α) versus time.
   - Differentiate α with respect to time to get $d\alpha/dt$

- Plot $d\alpha/dt$ vs. α and then fit equation 4.2 to the data using the curve fitting routine in kaliedagraph. Refer to Figure 4.3 for an example of a curve fit.

4. After steps 1-3 were repeated for all the data, Excel was again opened and plots were made for k, n, α_{final} (from equation 4.2 and found in step 3) versus temperature. The trendline function in Excel was used to fit exponential curves to each of those parameters. An average value was found for "m".

5. Again, another Excel spreadsheet was opened and time, α and $d\alpha/dt$ data for all isothermal runs were copied into one sheet. Excel was then used to predict α and $d\alpha/dt$ using equation 4.2 and the necessary coefficients as found via steps 3 and 4. Next, error was calculated between the measured and predicted α values. A final fitting of equation 4.2 was required as curve fitting in steps 3 and 4 were for individual curves and therefore did not consider everything together. Global minimization of error was performed using the Excel Solver routine. Final results of this analysis are presented in Chapter 5.

4.2.2 Resin Degradation Temperature

Resin degradation temperature is an important input to the optimizer. This parameter was used to prevent the optimizer from choosing cure cycles that resulted in degradation of the composite during curing. Resin degradation tests were performed

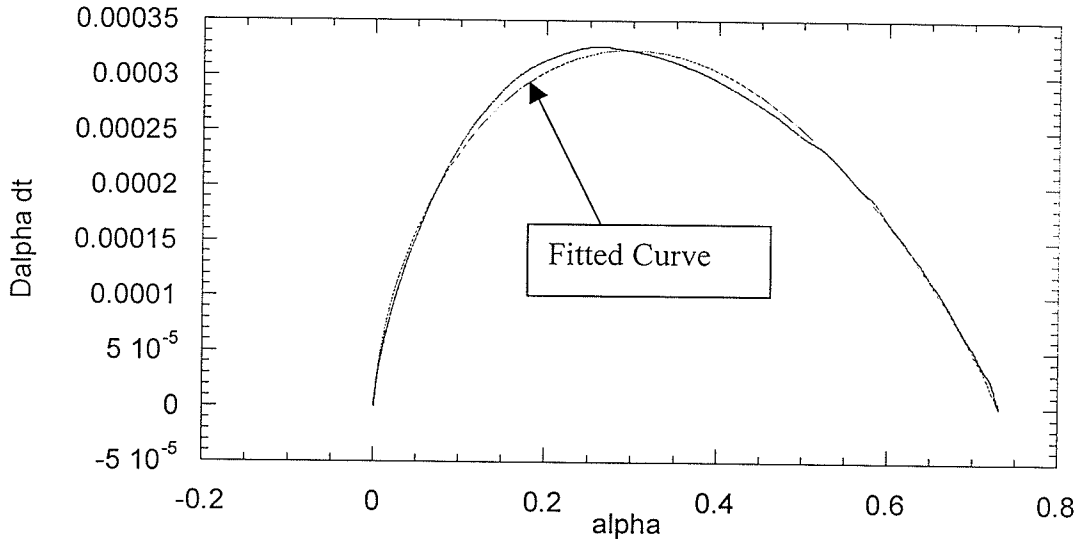

Figure 4.3: Example Plot of an Isothermal Run at 100°C – Plot of Measured and Fitted $d\alpha/dt$ versus α

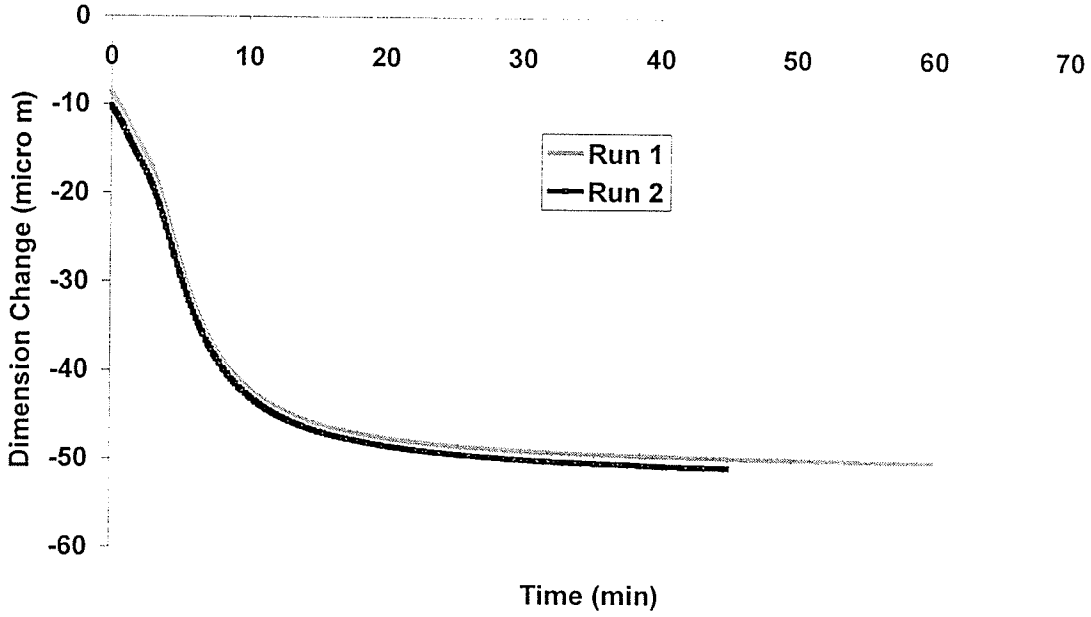

Figure 4.4: Plot of TMA Raw Data – Dimension Change (μm) versus time (min)

using the same equipment and sample set-up as was used for cure kinetics measurements. Three samples were subjected to dynamic scans at 2°C/min. This scan rate was chosen as faster dynamic rates delay the degradation point to higher temperatures. A low scan rate was used to obtain the lowest possible T_{deg}.

4.2.3 Cure Shrinkage

Cure shrinkage was used by the process model while calculating residual stresses and warpage. Cure shrinkage was measured through-the-thickness of the composite. Two samples were prepared which were 8 plies thick of uncured plain weave prepreg and approximately 7mm x 7 mm in size. Tests were preformed at Boeing Canada Technology – Winnipeg Division using a TA Instruments 2940 TMA (Thermal Mechanical Analyzer). Tests were performed at an isothermal temperature of 130°C for 1 hour. The macro expansion probe of 0.25" diameter with a static force of 5gm was used for the test. The test was performed by preheating the cell to 130°C at which time the cell was opened to insert the sample. After insertion, the cell was closed and allowed to re-equilibrated for 5 minutes.

Data from the TMA was in the form of dimensional change versus time as seen in Figure 4.4. To analyze the data, the dimensional change was converted into percent shrinkage by dividing the dimensional change by the original thickness. Next, the time and temperature data was applied to the kinetics model in order to plot the percent

shrinkage versus degree of cure. Finally, the data was modeled using a fifth order polynomial curve.

4.2.4 Coefficient of Thermal Expansion (CTE)

The process model requires CTE while calculating residual stresses and warpage. These testes were performed using fully cured samples that were approximately 7mm x 7mm in size and approximately 4mm thick. Tests were performed for the through-the-thickness direction at a dynamic scan rate of 2°C/min from room temperature to 130°C/min. Tests were performed on the TMA used for cure shrinkage measurements. A normal expansion probe of 0.1" diameter with 5gm static force was used for the experiments. Data from the TMA included dimensional change, temperature and time. Analysis was performed with the following steps:

1. The raw text data files were open using Excel. Temperature, dimensional change and original sample thickness were extracted for all runs and input into another Excel sheet.

2. Strain was calculated by dividing the change in thickness by the original thickness. The strain was then plotted versus temperature. From this plot, it was found that CTE was not linear and so a further step was required.

3. A third order polynomial curve was fit to the strain versus temperature data. This third order function was then differentiated to provide CTE as a function of temperature.

4.2.5 Specific Heat Capacity

Specific heat capacity was required for the thermochemical module of the process model. Specific heat capacity tests were performed in a similar manner to the cure kinetics tests. Tests were performed using a TA Instruments 2910 MDSC (Modulated Differential Scanning Calorimeter). Samples were fully cured and were prepared in the same manner as the cure kinetics samples i.e. 10-15mg in a non-hermetic aluminum pan. Three tests were performed using a dynamic scan rate of 2°C/min with a modulated signal of 0.1°C/min. Data from the MDSC was provided as Cp versus temperature. Therefore, the only analysis that was required was to fit a linear curve to the data.

4.2.6 Viscosity

Viscosity data was required by the process model to predict resin flow in the flow module. Tests were performed using a Bohlin CVO 120 Rheometer. Tests were performed on neat resin using 25mm parallel plates with a shear stress of 1 Pa and a frequency of 1Hz. The uncured resin samples were subjected to a variety of dynamic scanning rates (2, 3 and 5°C/min). Three tests were performed for 2°C/min while 2 samples were subjected to 3°C/min and 2 more to 5°C/min. Tests were performed from

room temperature until the point at which the resin gelled. The viscosity data was then modeled to equation 4.3:

$$\mu = A_\mu * \exp\left(\frac{-E_\mu}{RT}\right) * \left(\frac{\alpha_G}{\alpha_G - \alpha}\right)^{(A+B*\alpha)} \qquad (4.3)$$

A_μ is in Pas

R is the universal gas constant

T is temperature in Kelvin

E_μ is in kJ/mol

α_G is dimensionless

A is dimensionless

B is dimensionless

α is calculated from cure kinetics

Analysis to fit the data to the model required the following steps:

1. Time, temperature, and viscosity data from all runs were extracted from the data files and imported into one Excel spreadsheet. Using the time and temperature data and the cure kinetics model, the degree of cure was predicted for each point in time. This data was then plotted as viscosity versus degree of cure.

2. Initial viscosity predictions were made based on initial guesses of the various parameters in equation 4.3. Error was calculated between the actual and predicted values and error was summed.

3. The Excel Solver routine was then used to determine the parameters in equation 4.3 such that the error was minimized.

4.2.7 Gel Point

Gel point tests were performed using a Bohlin CVO 120 Rheometer. Samples were 4 plies of unidirectional uncured prepreg. Tests were performed at 2°C/min using 25mm parallel plates. The shear stress was set for 800 Pa at a rate of 1Hz. The gel time was defined as the time at which the elastic modulus and viscous modulus intersect. Information from the rheometer was in the form of elastic and viscous moduli versus time. Since the process model requires the degree of cure corresponding to the gel point, the temperature at which gelation occurred was compared to a MDSC run of 2°C/min.

4.3 Manufacturing of Angle Laminates

Along with experiments for material properties, experiments were performed to determine the temperature profile within an actual part as it cured in an autoclave. These experiments were used to confirm the heat transfer boundary conditions used in the process model.

To ensure proper heat transfer predictions, angle laminates were prepared at Boeing Canada Technology – Winnipeg Division as part of another study. The thermocouple data from these experiments was used to ensure the process model accurately predicted the temperature profile within the part. The tool/part configuration, called "Tool C" is shown in Figure 4.5 and Figure 4.6. Thermocouples were placed at the midplane of the laminate at the locations indicated in Figure 4.7. The tool was made of solid aluminum while the part was made of Cytec-Fiberite 934 resin with plain weave carbon fiber. The lay-up sequence for the laminate was $[0/90/45/-45/0/90/45/-45]_s$.

4.4 Simulation Details

The following sections discusses all the inputs required to run the process model and optimizer.

4.4.1 Process Model Inputs

There are three main inputs required for the process model. They are:

- Finite element Mesh
- Material Properties
- Boundary Conditions

Figure 4.8 shows the symmetric finite element mesh of Tool C. The symmetry of the part was exploited in order to speed up the computational time. The large blue area is

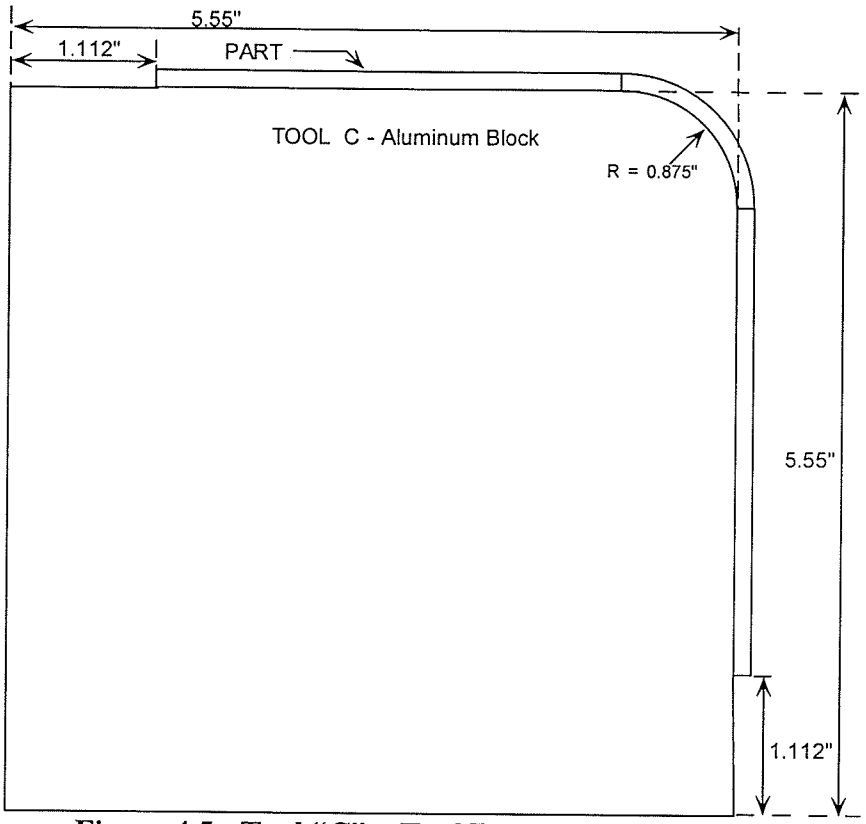

Figure 4.5: Tool "C" – Tool/Part used in Optimization Study

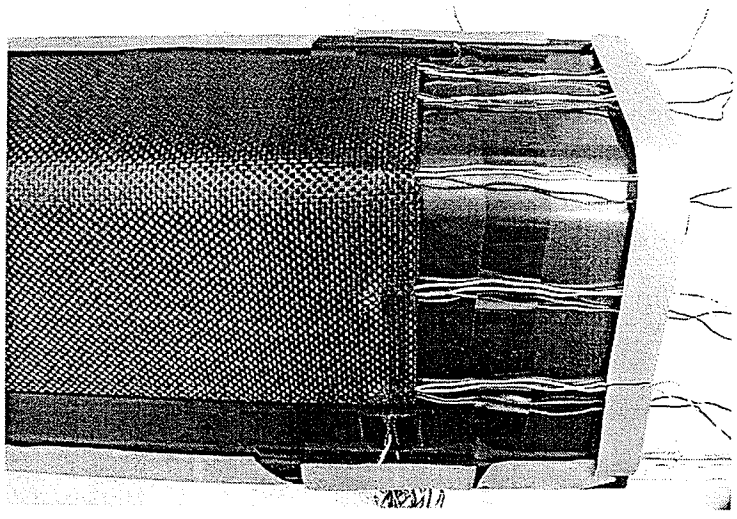

Figure 4.6: Picture of "Tool C"

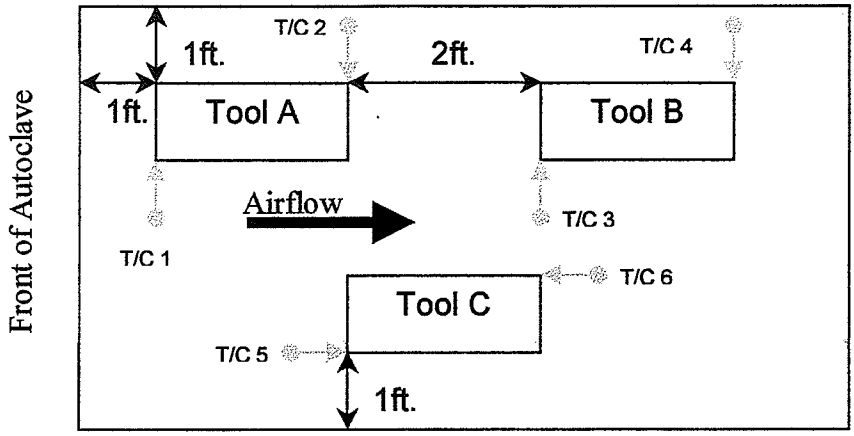

Figure 4.7: Autoclave Loading Arrangements for Angle Laminates

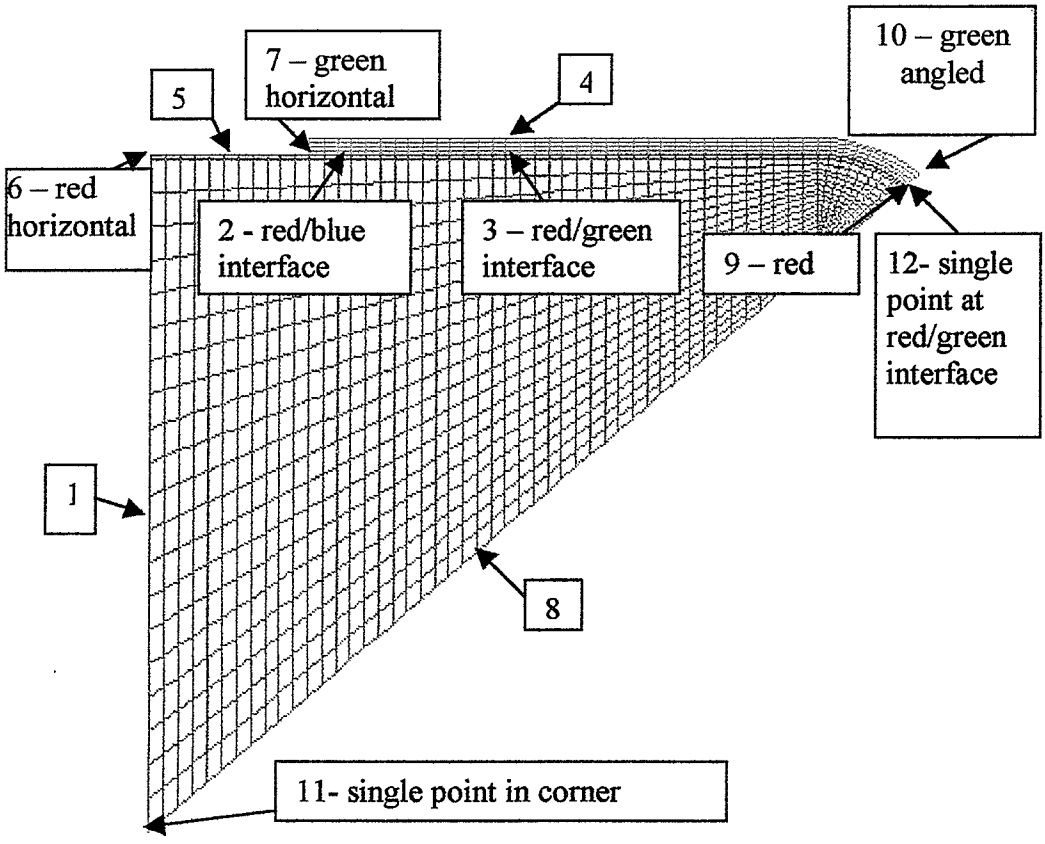

Figure 4.8: Boundary Condition Numbers on the Finite Element Mesh

the tool made of 6061 aluminum. Material properties for the aluminum were supplied by the process model but could be taken from any materials handbook. The green area is the composite. Composite used in this study was Cytec-Fiberite 934 resin with plain weave carbon fiber. Material properties for this material are summarized in Table 4.1. The model simulated each layer of carbon fiber with 4 "quarter" plies of unidirectional material that were laid up as $[0/90]_s$. The overall lay-up sequence for the laminate $[0/90/45/-45/0/90/45/-45]_s$. Red areas of the mesh represent the shear layer as was described in chapter 3. When the stress and flow modules were turned off, this material type was changed to 6061 Aluminum as the shear layer is only required for stress calculations.

Figure 4.8 gives the boundary numbers on the finite element mesh while Table 4.2 shows the boundary conditions associated with each boundary. A warpage study was performed before running the optimizer in order to validate the process model. The study used "Tool C" and was run for 4 different cure cycles with two different thermal cycles (cycles "1" and "2") as shown in Figure 4.9 and 4.10. For the study, the process model was used to predict springback angles by using a shear layer with a modulus of $1x10^6$ Pa. A springback angle is defined to be the angle by which the right angle laminated (Tool C) deforms after it is removed from the tool. A schematic of this type of deformation is pictured in Figure 3.5 The results from the models were compared to measured angles on laminates made at Boeing Canada Technology –Winnipeg Division.

Classification	Properties	Measured Values
Physical	Density Fiber Volume Fraction	Resin: 1300, Fiber: 1760 57.3 %
Cure Kinetics (general nth order model)	Total heat of reaction (kJ/kg oC) Activation Energy (kJ/mol) Log Z (1/min) Reaction order (n)	Hr = 483.1 E = 34.88 3.11 $n = n_0 + A/(1+\exp(-(T-T_0)/B))$ n_0 = 14.953, A = -13.244, T_0 = 139.218, and B = 9.951
Thermo-Physical	Specific heat capacity (J/kg oC) Thermal conductivity(W/m2 K) Cure Shrinkage CTE of Composite x 10-6 / oC	Resin: Cp = 2.7 T + 200.0 Fiber: Cp = 1.1 T + 1560.0 $K_{11} = 5.27 + 0.0389\ T$ $K_{22} = K_{33} = 2.567 + 0.0123\ T$ csc22 = csc33 = -4.1882*α + 1.3005 csc11 = -6.924x10-4*α + 9.122x10-6 $\alpha_{33} = \alpha_{22} = 36.5$ $\alpha_{11} = 3.5$
Rheological	Viscosity Gel Point Glass Transition Temperature (T_g)	$\mu = \mu_\infty \exp(k\alpha) \exp(U/RT)$ μ_∞= 8.858E-16 Pas, k = 28.264 U = 103.309 kJ/mol T = 177.4 °C, α = 0.4 $T_g = 19.538\exp(2.4617\alpha)$
Mechanical	Modulus of Composite GPa	$E_{11} = 38.571\alpha + 87.963$ $E_{22} = 2.448\alpha + 3.624$ $G_{12} = 0.1125\alpha - 0.0126$, $0.32 \le \alpha \le 0.86$ $G12 = 31.7\alpha - 27.1$, $0.86 \le \alpha \le 1.0$

Table 4.1: Material Properties for Cytec-Fiberite 934 Resin and Torray T 300 Fibers

Bndry	Ht Tans	Flow BC	Flow Mech BC	Stress BC	Tool Removal BC
1	Convection	impermeable	Free	Free	Free
2	adiabatic	impermeable	Free	Free	Free
3	adiabatic	impermeable	Sliding	Free	Free
4	Convection	impermeable	Prescribed pressure	Pressurized	Free
5	convection	impermeable	Free	Free	Free
6	Convection	impermeable	Free	Free	Free
7	Convection	Prescribed press	Prescribed pressure	Pressurized	Free
8	adiabatic	impermeable	Sliding	Sliding	Sliding
9	adiabatic	impermeable	Sliding	Sliding	Sliding
10	adiabatic	impermeable	Sliding	Sliding	Sliding
11	adiabatic	impermeable	Free	Fixed	Sliding
12	adiabatic	impermeable	fixed	Sliding	Fixed

Table 4.2: Boundary Conditions on Tool C

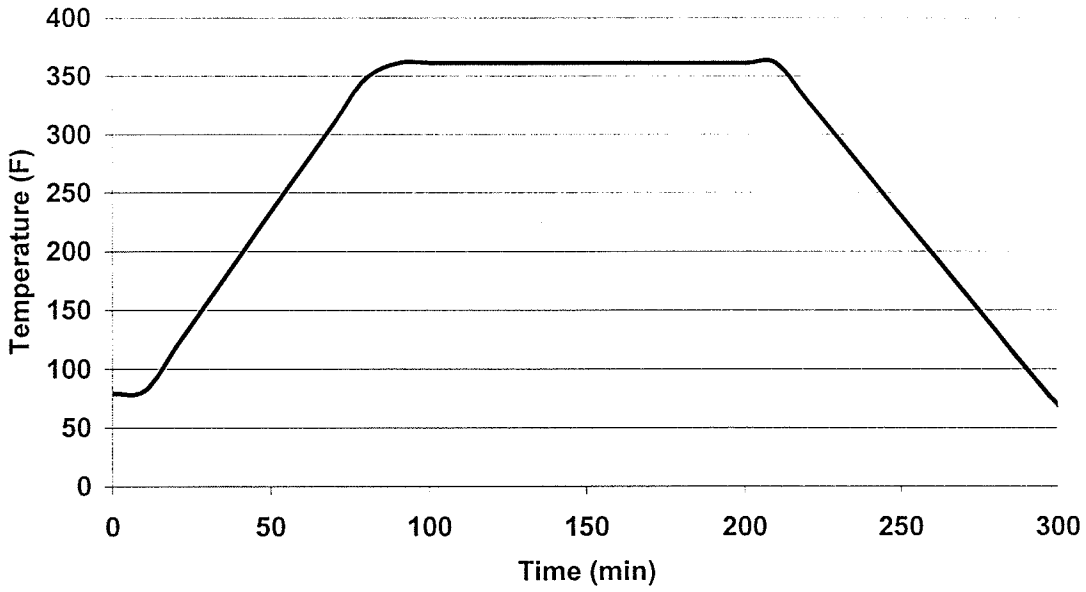

Figure 4.9: Cycle 1 Used for Warpage Study

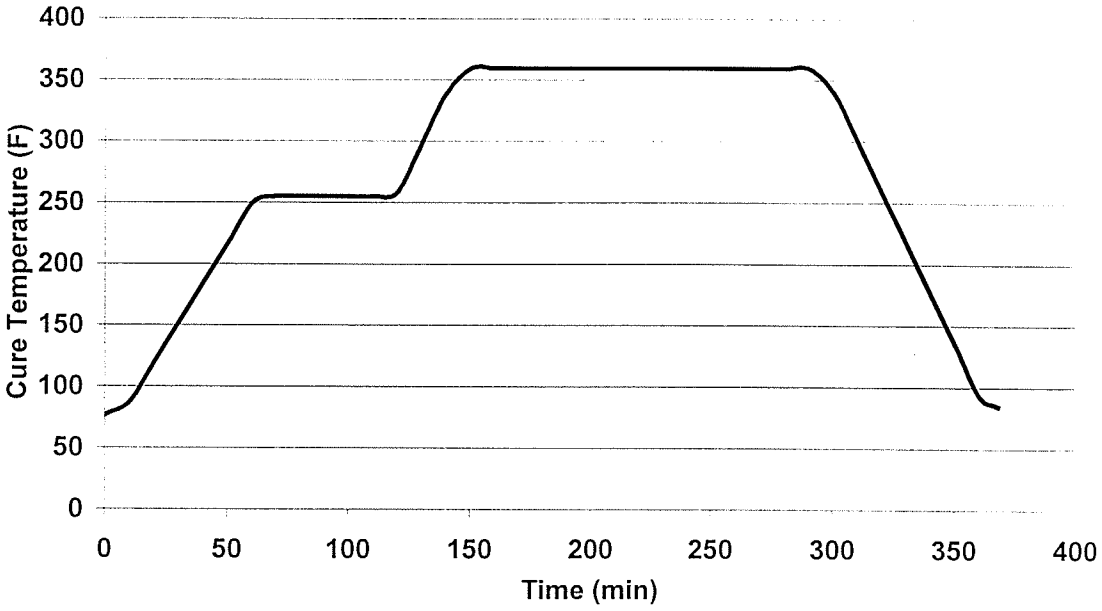

Figure 4.10: Cycle 2 Used in Warpage Study

Results of the warpage study are shown in Table 4.3. These results show that the process model fails to accurately predict the angles. It was for this reason that the stress and flow modules were turned off during optimization runs.

4.4.2 SIMCLAVE Input Details

Heat transfer "C" values, as required for Equation 3.13 and shown in Table 4.3, were taken from Hudek (9). These heat transfer values were used as the same autoclave that was used for the heat transfer modelling, was also used to make real parts from "Tool C". Note that the "C" values for the plain aluminum face (boundaries 1, 5 and 6) are higher than those for the sides covered with composite (boundaries 4 and 7). This was because the breather cloth and bagging material that is typically placed over prepreg during curing was insulating the composite.

4.4.3 Optimizer Input Details

Two input files, QAMeas.txt and GAINPT.dat are required to run the optimizer. QAMeas.txt is used by the Objective Function Module (OFM) for evaluating the fitness equation and GAINPT.dat is used to set-up GA options. A representative QAMeas.dat file is given in Table 4.5. Note that all the parameters are user definable. Parameters that are in bold and italics are the only ones used for the optimization runs by the OFM. All other parameters apply to parts of the fitness equation that were turned off (warpage, temperature gradient and cure gradient). Again, this was because warpage predictions of warpage by the process model that was being used were not satisfactory. Gradient

Pressure in Autoclave	Cycle	Measured Angle	Predicted Angle
85	1	1.38	1.844
45	1	1.69	1.8445
85	2	1.94	1.8819
45	2	1.26	1.8284

Table 4.3: Measured and Predicted Springback Angles for "Tool C"

Boundary Number	Effective "C" Values as found in Equation 3.13
1	0.318
4	0.119
5	0.318
6	0.318
7	0.119

Table 4.4: Effective "C" Values for Convective Boundaries for Finite Element Model of "Tool C"

FitnessPerfect		*1*
FitnessAcceptable		*0.75*
AlphaMin		*0.92*
TempCure		*355*
TempDeg		*390*
TempMax		*370*
TempRoom		*70*
Node1		1107
Node2		171
Node3		5
Node4		1
Postol		0.01
Negtol		0.01
tManfac		*310*
tOver		*20*
ThetaIdeal	90	
ThetaMax		92
ThetaMin		88
ThetaExtra	1	
Choice		1
DelTempMax	15	
DelTempLimit		10
DelAlphaMax		0.07
DelAlphaLimit		0.05
FTime		*1*
FTemp		*1*
FWarpage		*0*
FAlpha		*1*
FDeltAlpha	*0*	
FDeltTemp		*0*

Table 4.5: Sample of QAMeas.txt file

components (cure and temperature), as will be discussed in chapter 5, were turned off as it was found that cure and thermal gradients were within an acceptable limit for all optimization runs. The values in QAMeas.dat are material dependant and since there was only one material used during optimization runs, none of the values were changed. Note that temperature is in degrees F, time is in minutes, and for the fitness functions (Ftime, Ftemp, Fwarpage, Falpha, FdeltaAlpha, and FdeltaTemp) a value of 1 corresponds to "on" position and 0 corresponds to "off" position.

A representative GAINPT.dat file is given in Table 4.6. All options in GAINPT.dat are required for optimization unlike the QAMeas.txt file where the options are dependant on which fitness functions are "on". Parameters that are in bold and italics were changed during optimization runs and are the focus of the next section. It should be noted that the GA operates in regular GA mode when the number of micro-GA cycles (NMGA) is 1.

4.4.4 Optimization Test Matrix

Optimization runs were designed based on two criteria. The first criterion was to determine the best settings for the optimizer to be used for future work. The second criteria was to understand the effects of the following on the optimizer:

- Convergence – number of iterations until the solution converges
- GA mode versus micro-GA mode
- Effects of changing inputs in GAINPT.dat

```
'Number of members in a population NPOP ..... ', 10
'Number of generation NGEN ..... ', 40
'Number of micro-GA cycles NMGA ..... ', 10
'Apply fitness normalization? (y/n) ..... ', 'n'
'Probability of cross-over PCROSS ..... ', 0.8
'Probability of mutation PMUT ..... ', 0.9
'Allowable X-variations (minutes) XLIM ..... ', 15.
'Allowable Y-variations (deg F) YLIM ..... ', 15.
'Random number seed ..... ', 3
'Number of segments in a curing process NSEG ..... ', 7
'Maximum time for curing (minutes) TIMEMAX ..... ', 310.
'Min allow temp. where process terminates TMIN (deg F) ..... ', 80.
'Max autoclave air temperature TMAX (deg F) ..... ', 430.
'Initial temperature T0 (deg F) ..... ', 80.
'Maximum heating rate (positive, deg F per minute) ..... ', 18
'Maximum cooling rate (negative, deg F per minute) ..... ', -18.
'Location of the fitness output file from Fgen C++ routine .. ',
"C:\APCS\OUTPUTS\fittotal.out"
'Location of the fgen module..... ',
"C:\APCS\fgen\wholeproject.exe"
```

Table 4.6: Sample GAINPT.dat File

- Effects of "injecting" input cycles
- Effects of changing shapes of the fitness functions

The following sections discuss the optimization runs that were performed to study the above mentioned parameters. Please refer to Table 4.7, located at the end of the chapter, for a summary description of all optimization runs.

4.4.4.1 Convergence

There is a point during optimization at which time the fitness slows or ceases to increase. This is the point when the optimizer converges on the "optimal" solution. Runs 1, 3 and 4 looked at the time required to converge. Run 1 used the regular GA for a total of 2000 populations. Run 1 was sufficiently long to make a first estimate of the number of populations required for convergence. Runs 3 and 4 were used to verify that the number of populations chosen from run number 1 would work for both the regular and micro GA modes.

4.4.4.2 GA versus Micro-GA

It has been shown in other studies that micro-GAs optimize towards more optimal solutions (20). In order to study the differences between running a particular configuration in regular GA mode and micro-GA modes, several of the runs as described in these sections were run in both modes. Runs 3, 5, 7, 9, and 11 (regular GA mode) were all duplicated in micro–GA mode (Runs 4, 6, 8, 10, and 12).

4.4.4.3 Effect of Inputs in GAINPT.dat File

The settings of the GA have a large impact on the quality of an optimization run. These settings are found in the GAINPT.dat file. Only options in the file that have been changed during at least one test are shown in Table 4.6. All other values were kept constant and were held at the values as seen in Tables 4.3 and 4.4. The list of variables in GAINPT.dat that were studied directly or indirectly are:

- NGEN (Number of Generations)
- NMGA (Number of Micro-Generations)
- PMUT (Probability of Mutation)
- Random Seed Number
- Max HR (Maximum Heating rate – positive ramp rate)

Runs 2, 16 and 17 were used to directly study the random seed number, PMUT, and NMGA respectively. Run 2 studied the effects of the random seed number on the quality of the solution. Runs 16 and 17 were variations of Run 12 which is discussed in the following section. Runs 16 used the same settings as Run 12 except the probability of mutation (PMUT) was changed from 0.9 to 0.1. Run 17 was again similar to Run 12 except the number of micro-cycles to be run was changed. Other GAINPT.dat parameters such as maximum heating rate and number of total populations were studied indirectly in other studies in sections 4.4.3.1 and 4.4.3.4 respectively.

4.4.4.4 Influence of Input Cycles

Initial optimization runs (Runs 1-4) all exceeded the maximum allowable temperature (TempMax) as set in QAMeas.dat. Cycles were required to be input or "injected" because it was found that the random cycle generator could not produce an acceptable cycle given maximum heating ramp rates above 7.9°F/min. This is believed to be caused by the way in which the first set of random cycles are created. Initial cycles are created by taking the maximum allowable time and dividing it by the number of segments. Therefore, in the case of the current settings, maximum allowable time of 310 minutes and 7 segments, each segment is approximately 44 minutes. Once the cycle is divided up, a ramp rate is assigned to a segment by taking the maximum ramp rate, say 18°F/min (the maximum the autoclave can produce) and multiplying it by a random number between 0 and 1. This is done for each of the first 6 segments (the ramp up segments, the 7^{th} segment is the cool down segment). If the temperature at the end of the 6^{th} segment exceeds the maximum allowable temperature then the cycle fails and is assigned a very low fitness. It requires 6 small random numbers in a row to achieve this. For example, assume the first 6 ramp up segments had equal ramp rates. All 6 segments would require a ramp rate of less than 1.6°F/min in order to not exceed the maximum temperature restriction. In order to get a ramp rate of 1.6F/min or less, the random numbers would all have to be less than 0.09 (18F/min * 0.09 is approximately 1.6). Unfortunately, the random number generator would not come up with such a string of small numbers and so a high ramp rate cycle had to be injected.

The input cycles were inserted so that they replaced the first one or two random cycles (depending on the settings) that are usually generated by the GA. Both input cycles are shown in Figure 4.11. The first cycle to be injected was the manufacturer's recommended cure cycle and the second was a cure cycle with the maximum possible autoclave heating rate (18F/min). After initial Runs 1 through 4, Runs 5 and 6 were used to study the effects of injecting only the manufacturer cycle while Runs 7 and 8 had both cycles injected.

4.4.4.5 Influence of Shape of Fitness Functions

The next major area that was explored was the effect of the fitness functions. Three versions of the cure function and two versions of the time function were developed. Runs 1-8 use the first versions of both the cure and time fitness functions. The remaining runs (Runs 9-17) all use different combinations of these fitness functions as described below. The numbers in brackets are the function version number which are depicted in Figures 4.11 to 4.14.

Run 9: Cure (2), Time (1)

Run 10: Cure (2), Time (1)

Run 11: Cure (2), Time (1)

Run 12: Cure (2), Time (1)

Run 13: Cure (3), Time (1)

Run 14: Cure (2), Time (2)

Run 15: Cure (3), Time (2)

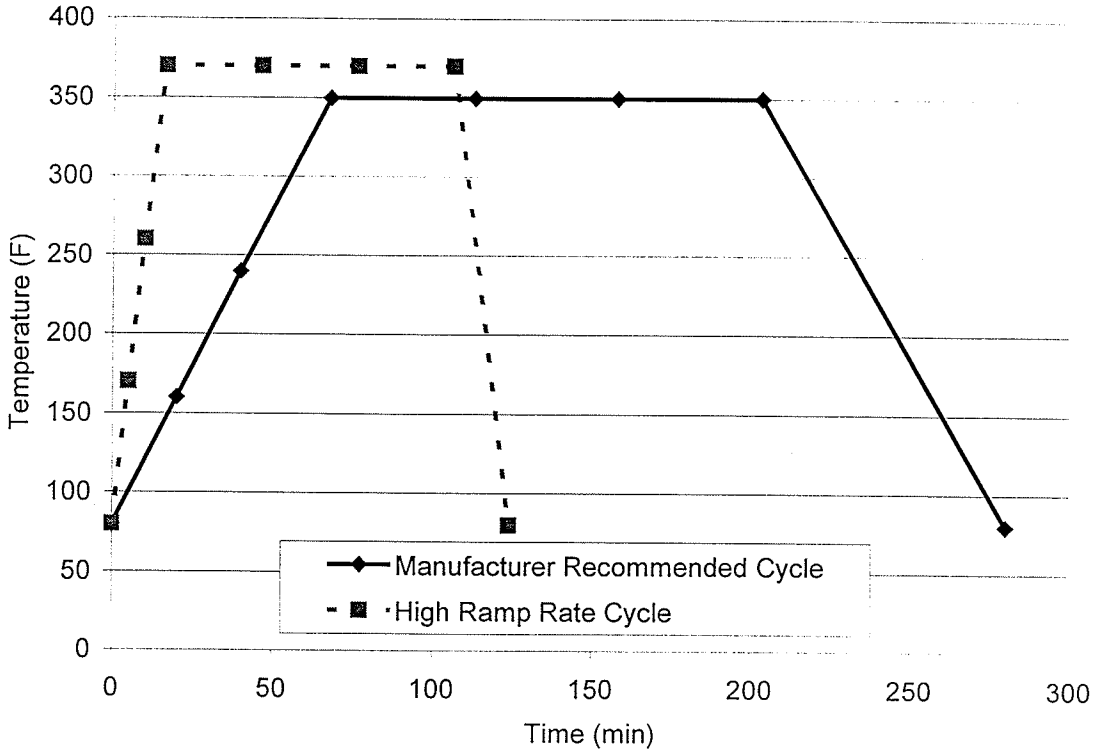

Figure 4.11: Plot of Manufacturer's Recommended Cure Cycle (for Cytec-Fiberite 934 Resin) and High Ramp Rate Cycle

The following gives the shape and equations for the various versions of the cure and time functions. Version 1 of the cure function has already been given in Section 3.1.2.1 and version 1 of the time function is in Section 3.1.2.4.

Cure fitness function version 2, as seen in Figure 4.12, was designed to favour the minimum degree of cure. The mathematical form for this version. is given in the following equations.

For $\alpha_{min} < \alpha < 1$:

$$f(\alpha_{final}) = a\ \alpha_{final} - b \qquad (4.4)$$

where a and b are constants.

The bounds for this relation are:

$f(\alpha_{final})$ = FitnessPerfect for $\alpha = \alpha_{min}$

$f(\alpha_{final})$ = FitnessAcceptable for $\alpha = 1$

For $\alpha < \alpha_{min}$

$$f(\alpha_{final}) = c \exp (d\ \alpha_{final}) \qquad (4.5)$$

where c and d are constants.

The bounds for this equation are:

$f(\alpha_{final})$ = FitnessPerfect for $\alpha = \alpha_{min}$

$f(\alpha_{final}) = 0.0001$ for $\alpha = 0$

With these assigned bounds, the values of constants a, b, c, and d are:

$a = (\text{FitnessAcceptable} - \text{FitnessPerfect}) / (1 - \alpha_{min})$

$b = a - \text{FitnessPerfect}$

$c = 0.0001$

$d = (1 / \alpha_{min}) * \ln (\text{FitnessPerfect} / c)$

Cure fitness function version 3, as seen in Figure 4.13, was designed so that all degrees of cure over the minimum would have the same fitness. The mathematical form for this version is given in the following equations.

For $\alpha_{min} < \alpha < 1$

$$f(\alpha_{final}) = 1 \qquad (4.6)$$

For $\alpha < \alpha_{min}$

$$f(\alpha_{final}) = a \exp (b\, \alpha_{final}) \qquad (4.7)$$

where a and b are constants

The bounds for this equation are:

$f(\alpha_{final}) = \text{FitnessPerfect}$ for $\alpha = \alpha_{min}$

$f(\alpha_{final}) = 0.0001$ for $\alpha = 0$

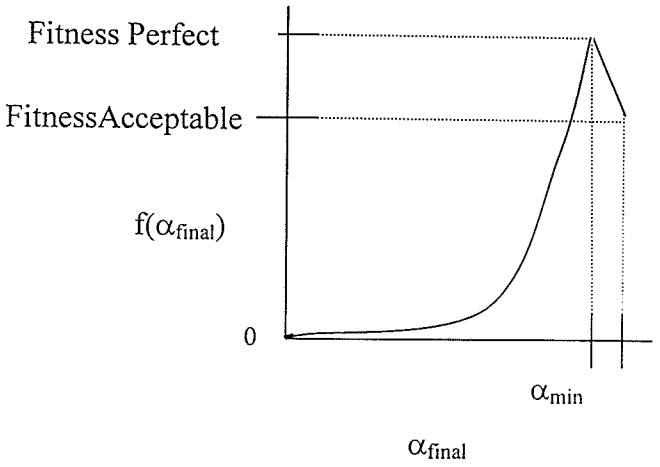

Figure 4.12: Shape of $f(\alpha_{final})$ versus Final Degree of Cure – Version 2

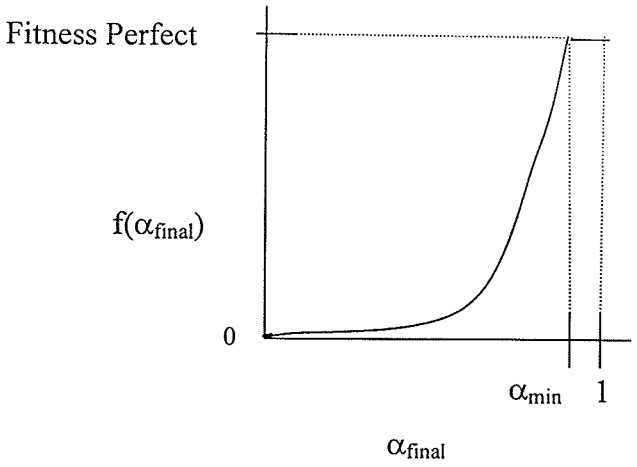

Figure 4.13: Shape of $f(\alpha_{final})$ versus Final Degree of Cure – Version 3

With these assigned bounds, the values of constants a and b are:

$a = 0.0001$

$b = (1 / \alpha_{min}) * \ln (FitnessPerfect / a)$

Time fitness function version 2, as seen in Figure 4.14, was designed to drive the optimizer towards cycles with shorter times by increasing the slope of the function as compared to the linear function used in version 1. The mathematical form for this version is given in the following equations.

For $t < t_{manufacturer}$,

$$f(t) = a \exp (b\ t) \qquad (4.8)$$

where a and b are constants

The bounds for this equation are:

$f(t) = FitnessPerfect$ for $t = t_{min}$

$f(t) = FitnessAcceptable$ for $t = t_{manufacturer}$

For $t > t_{manufacturer}$

$$f(t) = c \exp (d\ t) \qquad (4.9)$$

where c and d are constants

The bounds for this equation are:

$f(t) = FitnessAcceptable$ for $t = t_{manufacturer}$

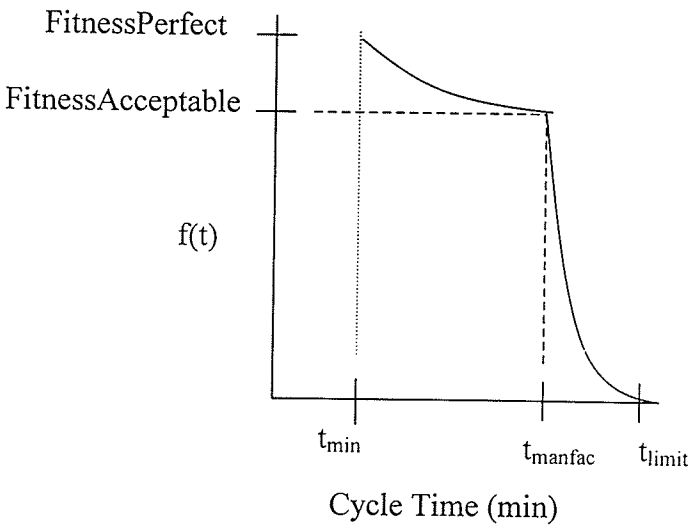

Figure 4.14: Shape of f(t) versus Cycle Time – Version 2

$f(t) = 0.0001$ for $t >> t_{manufacturer}$

With these assigned bounds, the values of constants a, b, c, and d are:

$b = [1/(t_{manfac} - t_{Min})] * \ln (FitnessAcceptable/FitnessPerfect)$

$a = FitnessPerfect * exp(-1*b* t_{Min})$

$d = [1/(t_{Limit} - t_{Manfac})] * \ln (0.0001 / FitnessAcceptable)$

$c = FitnessAcceptable * exp (-1* d *t_{Manfac})$

4.4.4.6 Summary

The previous sections described the optimization runs used to study the following areas:

- Convergence – number of iterations until the solution converges
- GA mode versus micro-GA mode
- Effects of changing inputs in GAINPT.dat
- Effects of "injecting" input cycles
- Effects of changing shapes of the fitness functions

Table 4.7 is provided as a master summary list of all optimization runs. Results are discussed in Section 5.

Run No.	Objective	NPOP	NGEN	NMGA	PMUT	Seed	Max HR (pos)	Input manfac cycle (y/n)	Input High Ramp Cycle	Change Fitness Function
1	Determine number of iterations to convergence	10	2000	1	0.9	3	7.9	n	n	n
2	Determine the effect of random seed	10	400	1	0.9	7	7.9	n	n	n
3	Set baseline of Regular GA	10	400	1	0.9	3	7.9	n	n	n
4	Set baseline of micro GA	10	40	10	0.9	3	7.9	n	n	n
5	Effects of injecting manufacturer cycle (Reg GA)	10	400	1	0.9	3	7.9	y	n	n
6	Effects of injecting manufacturer cycle (Micro GA)	10	40	10	0.9	3	7.9	y	n	n
7	Effects of injecting manufacturer cycle and high ramp rate cycle (Reg GA)	10	400	1	0.9	3	18	y	y	n
8	Effects of injecting manufacturer cycle and high ramp rate cycle (Micro GA)	10	40	10	0.9	3	18	y	y	n
9	Effects of injecting manufacturer cycle with new cure fitness function (version 2) (Reg GA)	10	400	1	0.9	3	7.9	y	n	y
10	Effects of injecting manufacturer cycle with new cure fitness function (version 2) (Micro GA)	10	40	10	0.9	3	7.9	y	n	y
11	Effects of injecting manufacturer cycle and high ramp	10	400	1	0.9	3	18	y	y	y

	rate cycle with new cure fitness function (version 2) (Reg GA)									
12	Effects of injecting manufacturer cycle and high ramp rate cycle with new cure fitness function (version 2) (Micro GA)	10	40	10	0.9	3	18	y	y	y
13	Effects of injecting manufacturer cycle and high ramp rate cycle with new cure fitness function (version 3) (Micro GA)	10	40	10	0.9	3	18	y	y	y
14	Effects of new time function (version 2) and 2 injected cycles with new cure function (version 2) (Micro GA)	10	40	10	0.9	3	18	y	y	y
15	Effects of new time function (version 2) and 2 injected cycles with new cure function (version 3) (Micro GA)	10	40	10	0.9	3	18	y	y	y
16	Effects of PMUT on setup as used in Run 12	10	40	10	0.1	3	18	y	y	y
17	Effect of changing NMGA on setup used in Run12	10	10	40	0.9	3	18	y	y	y
18	Optimal Settings – Cure function (Version 2), Time Function (Version 2), inject 2 cycles, with high NMGA in the Micro-GA mode	10	10	40	0.9	3	18	y	y	y

Table 4.7: Settings and Objectives for Optimization Runs

CHAPTER 5

Results and Discussion

5.0 Introduction

This chapter presents and discusses the results of material testing and cure cycle optimization runs as described in chapter 4. While representative graphs are provided here, the rest can be found in Appendices A and B.

5.1 Material Characterization Results

Test results for the following material properties are presented in this section.

- Cure Kinetics parameters
- Resin Degradation Temperature (T_{deg})
- Cure Shrinkage
- Coefficient of Thermal Expansion (CTE)
- Specific Heat Capacity
- Viscosity
- Gel point

5.1.1 Cure Kinetics

Figure 5.1 shows the data from a dynamic scan of 2°C /min. By integrating the area under this curve and taking the average of three trials, the total heat of reaction of the composite was found to be 128 J/g.

Following an analysis of the isothermal runs, as described in Section 4.2.1, the cure kinetics model (Equation 5.1) and model constants are listed below. Figure 5.2 plots experimental and predicted results of α versus $d\alpha/dt$ for an isothermal run at 110°C. Graphs for isothermal runs of 90, 100, 110, 120 and 130°C can be found in Figures A.1 to A.5 respectively.

$$\frac{d\alpha}{dt} = k * \alpha^m * (\alpha_{final} - \alpha)^n \qquad (5.1)$$

where:

k = Z exp (-E/RT)

Z = 1.502E+08 sec^{-1}

E = 78.809 kJ/mol

T is temperature in Kelvin

R is the universal gas constant

m = 0.63722

n = A exp (B*T)

A = 0.0013366

B = 0.0175435 K^{-1}

T is temperature in Kelvin

Alphafinal = C exp (D*T)

C = 0.0011557

D = 0.017402 K^{-1}

T is temperature in Kelvin

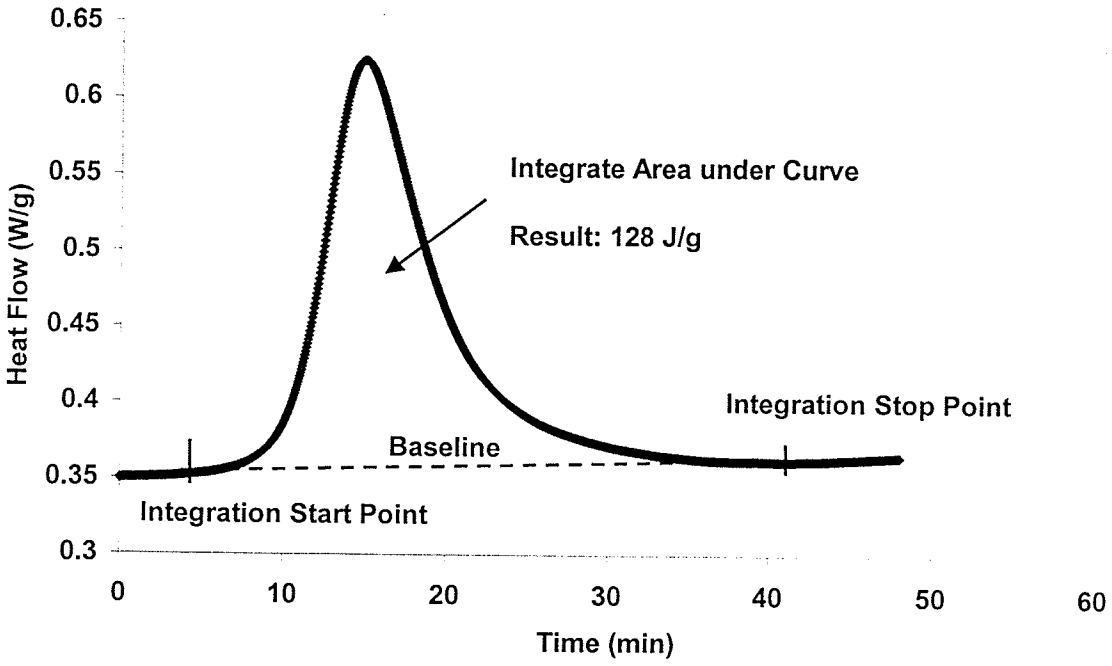

Figure 5.1: Plot of Heat Flow versus Time for a Dynamic Scan of 2°C /min

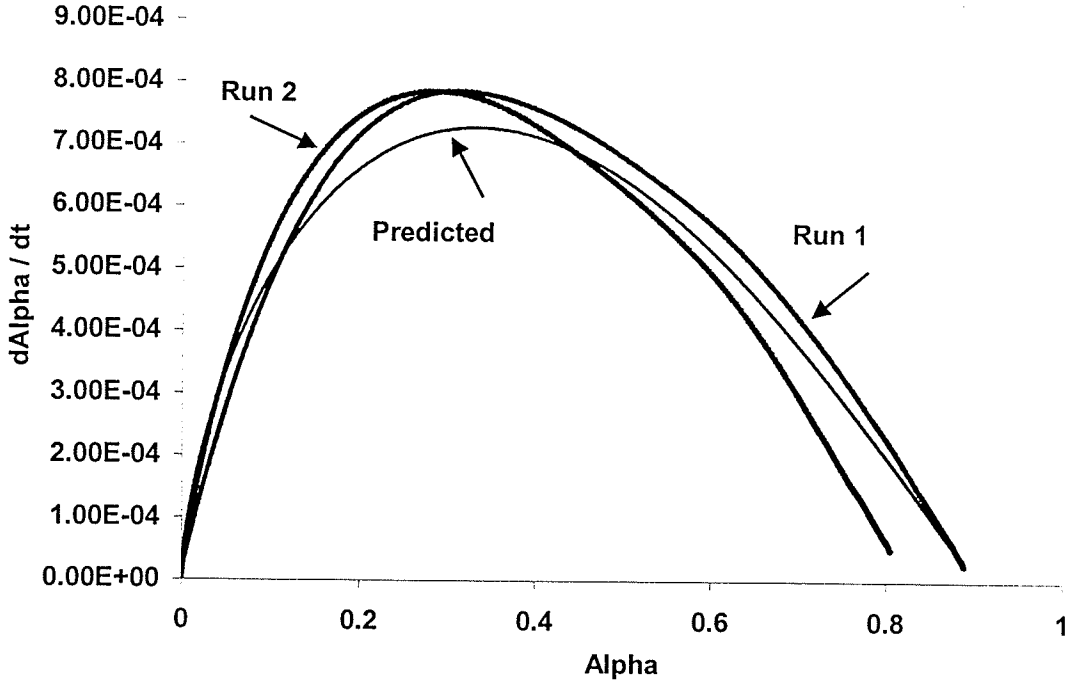

Figure 5.2: Plot of α versus $d\alpha/dt$ for Isothermal Run at 110°C

Figures A.1 to A.5 show a run-to-run variation in the $d\alpha/dt$ vs. α curves. The figures also show that the cure kinetics model tends to slightly underpredict the cure rate ($d\alpha/dt$). Run-to-run variation was caused primarily by the opening of the test cell to insert the test sample at a given isothermal temperature as described in Section 4.2.1. The time that the cell was opened varied from run to run and so caused a slight difference in the time it took for the cell to re-equilibration back to the isothermal temperature.

5.1.2 Resin Degradation Temperature

Figure 5.3 shows heat flow versus time for 3 dynamic scans at 2°C/min. The peaks on the left are caused by the kinetic reaction and are similar to the curve in Figure 5.1. The second set of peaks, on the right, are caused by the degradation of the resin. The resin degradation temperature is taken as the temperature at which the resin begins to degrade which corresponds to 250°C. Notice that the curves do not all line up on a horizontal plane. This was due to the difference in mass from sample to sample which causes the heat flow baseline to be different for each case.

5.1.3 Cure Shrinkage

Following analysis as described in Section 4.2.3, Figure 5.4 shows a plot of experimental and predicted values of percent cure shrinkage versus degree of cure. Predictions were made using the model shown in Equation 5.2.

$$\% \text{ shrinkage} = 18.0635\,\alpha^5 - 40.6545\,\alpha^4 + 25.026\,\alpha^3 - 2.50645\,\alpha^2 - 2.6813\alpha \qquad (5.2)$$

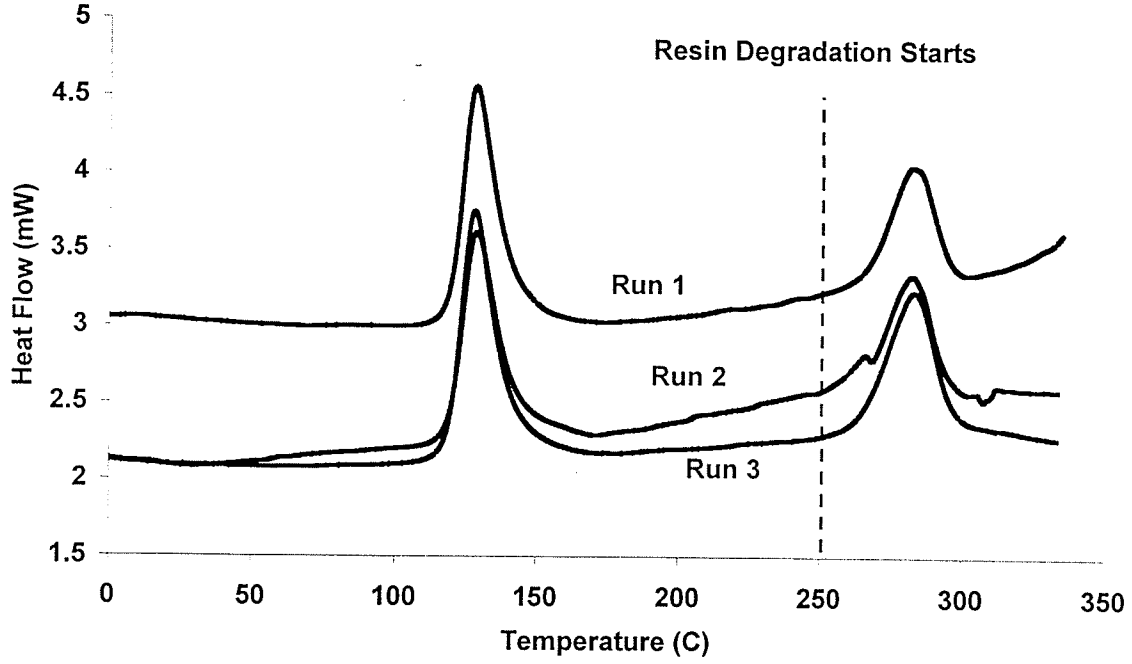

Figure 5.3: Resin Degradation Temperature of Hexcel F155 Resin

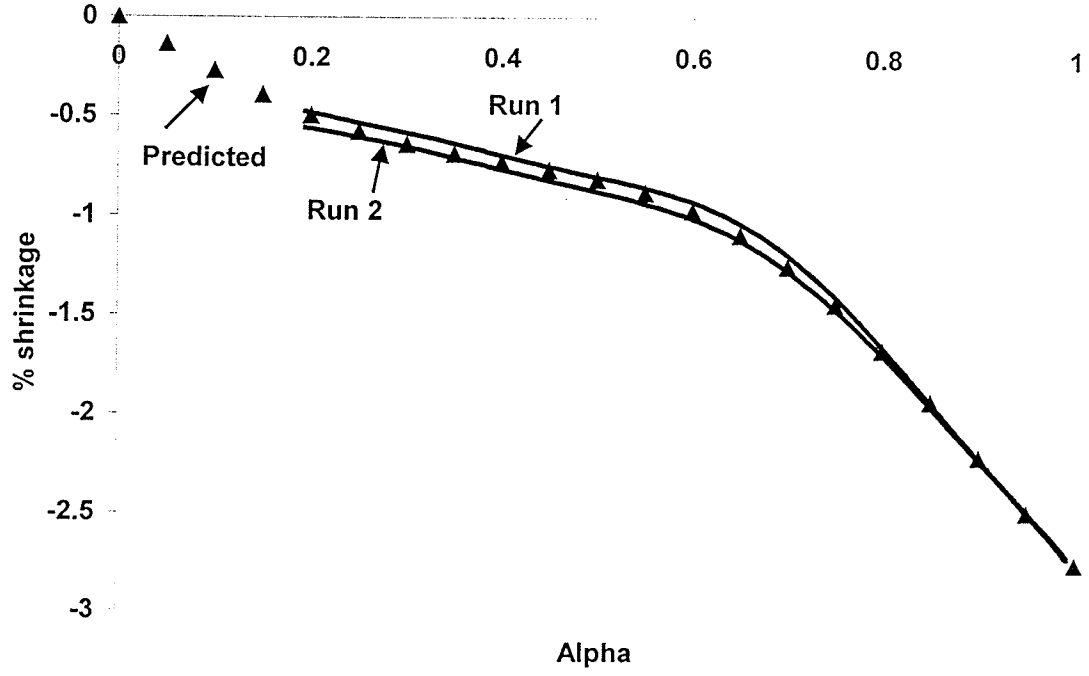

Figure 5.4: Percent Cure Shrinkage versus Degree of Cure

Notice in Figure 5.4 that the percent cure shrinkage starts only after a degree of cure of 0.2. This was caused by the 5 minutes the cell took to equilibrate after it had been opened to put the sample inside. The model used in equation 5.2 was fit to the data such that it accounted for this "gap" in the empirical data. The model was fit such that the percent shrinkage would be zero when the degree of cure was zero.

5.1.4 Coefficient of Thermal Expansion (CTE)

Data provided by the TMA, as plotted in Figure 4.4, was in the form of dimensional change (μm) versus time (min). As discussed in Section 4.2.1.4, strain (Eps in m/m) was calculated and potted versus temperature as shown in Figure 5.5. For some materials, the slope for such a plot is constant which means CTE is constant. However, as is seen in this plot, the slope is not constant. Therefore, Equation 5.3, was fit to the data which provided strain as a function of temperature. Equation 5.3 was then differentiated into Equation 5.4 to provide an equation that related CTE to temperature.

$$\text{Strain} = 5.590\text{E-}9*T^3 - 6.570\text{E-}7*T^2 + 9.439\text{E-}5*T - 2.14\text{E-}3 \qquad (5.3)$$

Where: strain is in (m/m)

T is in °C

$$\text{CTE} = 1.677\text{E-}8*T^2 - 1.314\text{E-}6*T + 9.439\text{E-}5 \qquad (5.4)$$

Where: CTE is in (m/m°C)

T is in °C

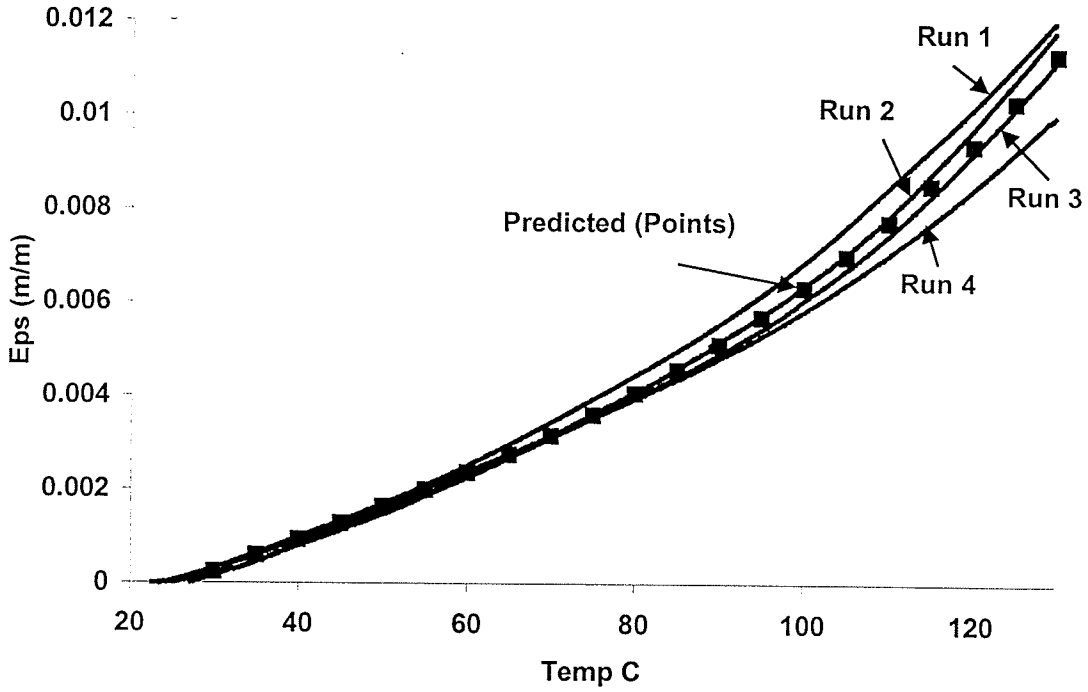

Figure 5.5: Plot of Experimental and Predicted Strain versus Temperature Data

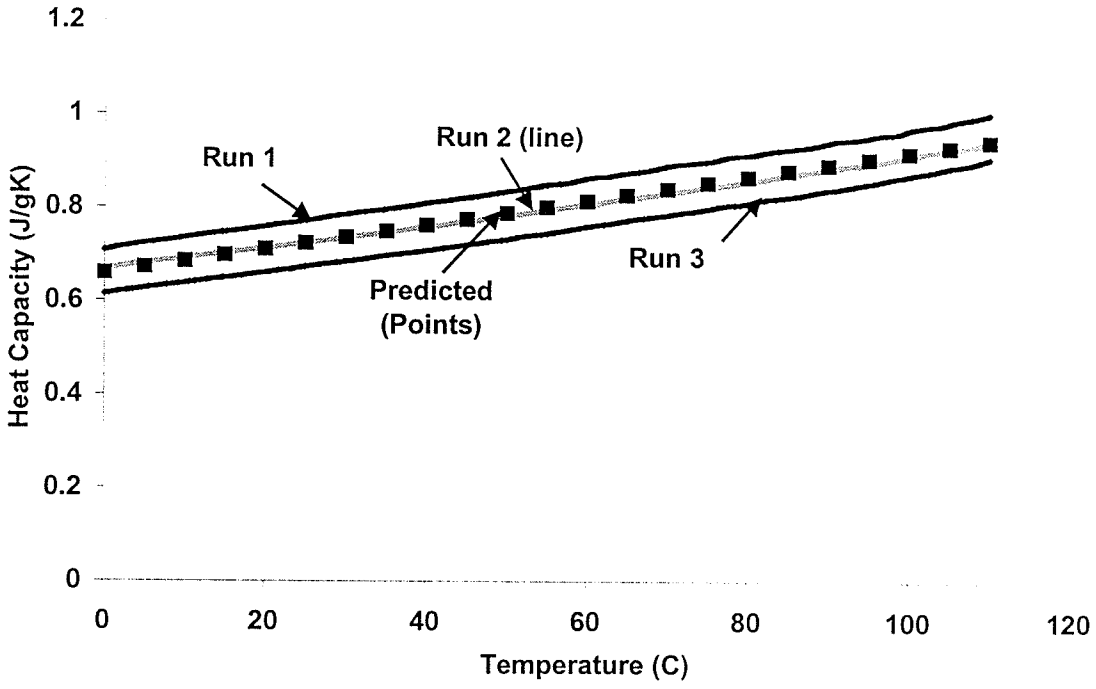

Figure 5.6: Plot of Experimental and Predicted Cp versus Temperature

5.1.5 Specific Heat Capacity (Cp)

The MDSC proved to be an efficient tool for determining specific heat capacity as data was supplied as specific heat capacity versus temperature. Therefore, only a linear curve had to be fit to the data as seen in Figure 5.6. The average fit of these curves gives Equation 5.5:

$$Cp = 0.6589 + 0.0025*T \qquad (5.5)$$

T is in °C

Cp is J/g K

5.1.6 Viscosity

Viscosity data from the rheometer was in the form of viscosity versus temperature. All dynamic scan data is plotted in Figure 5.7. As per analysis discussed in Section 4.2.6, the data was fit to Equation 5.6. Figures A.6 to A.8 show plots of experimental and predicted values of viscosity vs. alpha for dynamic scan rates of 2, 3, and 5°C/min respectively.

$$\mu = A_\mu * \exp\left(\frac{-E_\mu}{RT}\right) * \left(\frac{\alpha_G}{\alpha_G - \alpha}\right)^{(A+B*\alpha)} \qquad (5.6)$$

Where:

A_μ = 6.5452E-12 Pas

E_μ = 90477.8 J/mol

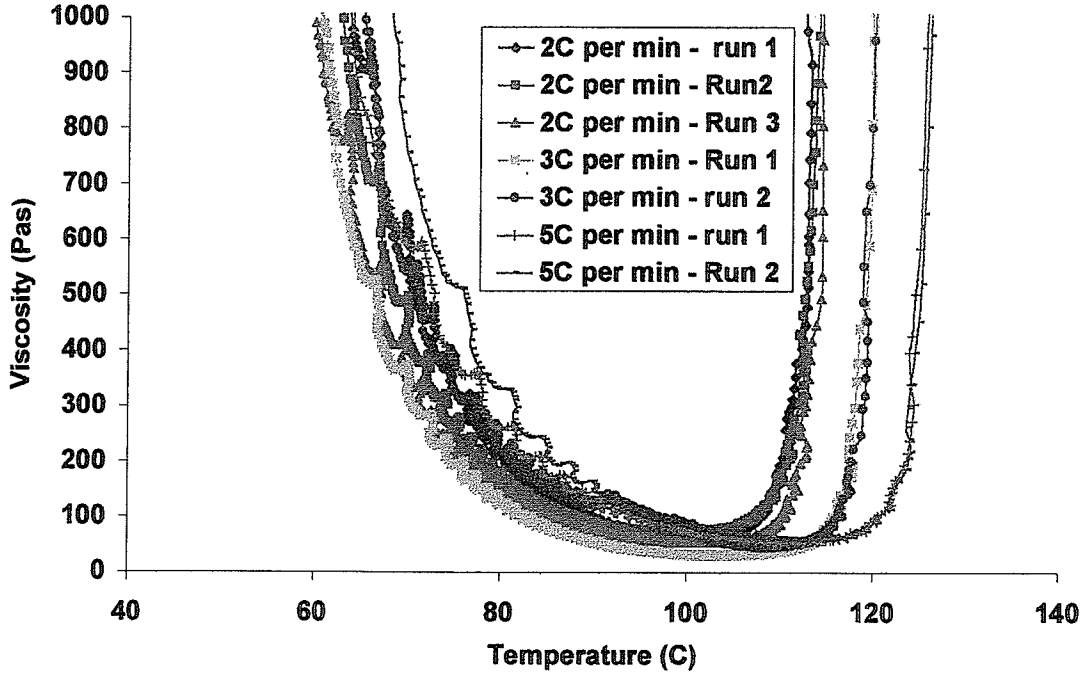

Figure 5.7: Plot of Viscosity versus Temperature for all Dynamic Scans

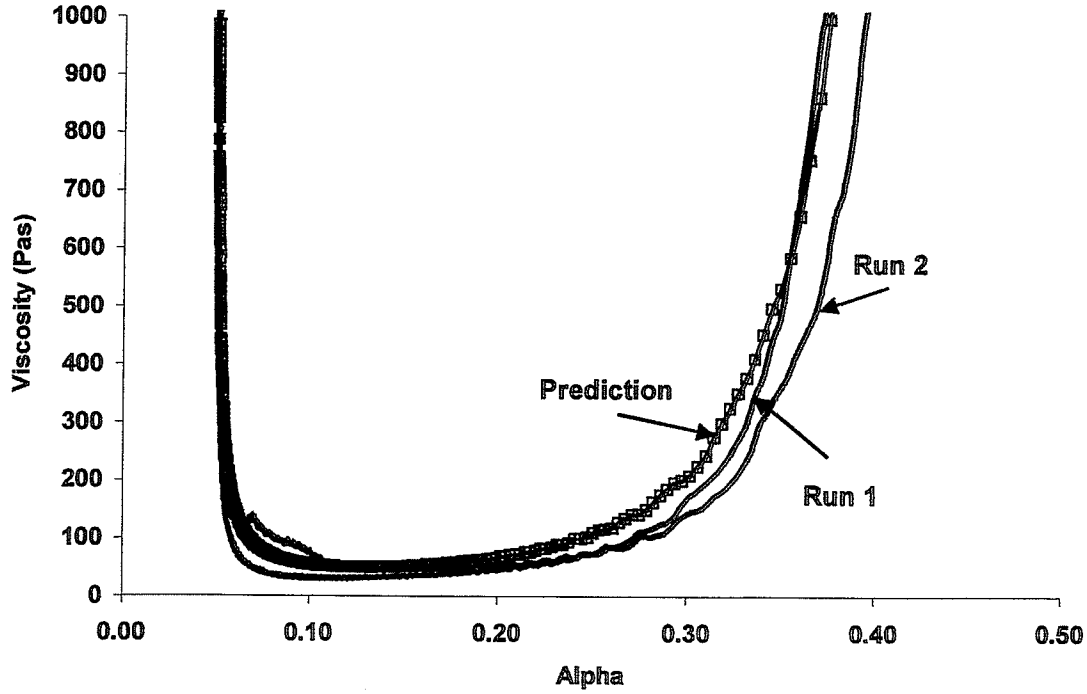

Figure 5.8: Plot of Experimental and Predicted Values for Viscosity versus α for a ramp rate of 2°C/min

R is the universal gas constant

T is in Kelvin

$\alpha_G = 0.533$

A = 3.185

B = 2.359

α is calculated from cure kinetics

From Figure 5.7, it can be seen that the viscosity decreases as the resin warms until such time that the resin cures enough that the viscosity begins to rise. Higher ramp rates appear to delay the increase of viscosity as seen in Figure 5.7. However, when the data is plotted versus degree of cure, as in Figures A.6 to A.8, it is shown that the increase in viscosity occurs at approximately the same degree of cure.

Figure 5.8 shows the experimental and predicted values for viscosity versus degree of cure for a ramp rate of 2°C/min. From this plot, it can be seen that the model predicts the value within the error of the experiments. However, the model does over predict for a ramp rate to 5°C/min, as shown in Figure A.8. This could be attributed to the method in which the rheometer heats the sample. The sample, which is between two plates, is contained within a test chamber that blows air across one side of the sample in a direction parallel to the plates. At higher ramp rates, the sample may not be heated uniformly which could cause the error between the model and experimental data in the 5°C/min dynamic scan.

5.1.7 Gel Point

Figure 5.9 plots the viscous and elastic modulus versus temperature for a dynamic scan of 2°C /min. This graph shows gelation occurs at 128.7°C at this particular scan rate. This temperature was then correlated with data from an MDSC run at the same scan rate as shown in Figure 5.10. By correlating the two graphs, gelation was found to occur at a degree of cure of 0.35.

5.1.8 Summary of Material Properties

Data from the above material characterization tests are summarized in Table 5.1.

5.2 Simulations Results

At the onset, process model predictions are validated using experimental part temperature measurements. Subsequently, the results from cure cycle optimization runs that were performed as outlined in Section 4.4.4, are presented and discussed. Individual results of each optimization run are given in Appendix B.

5.2.1 Validation of Process Model

Data from thermocouples 5 and 6, as shown in Figure 4.7, were used to verify the thermal boundary conditions and therefore the thermal predictions made by the process

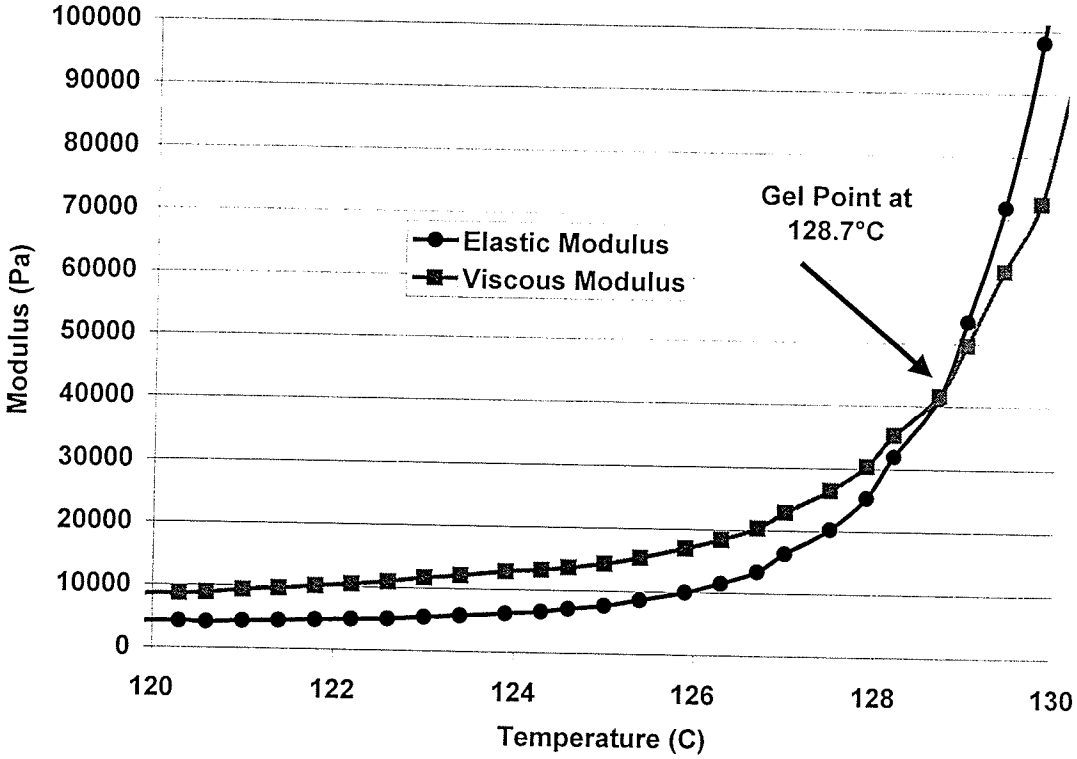

Figure 5.9: Plot of Elastic and Viscous Modulus versus Temperature

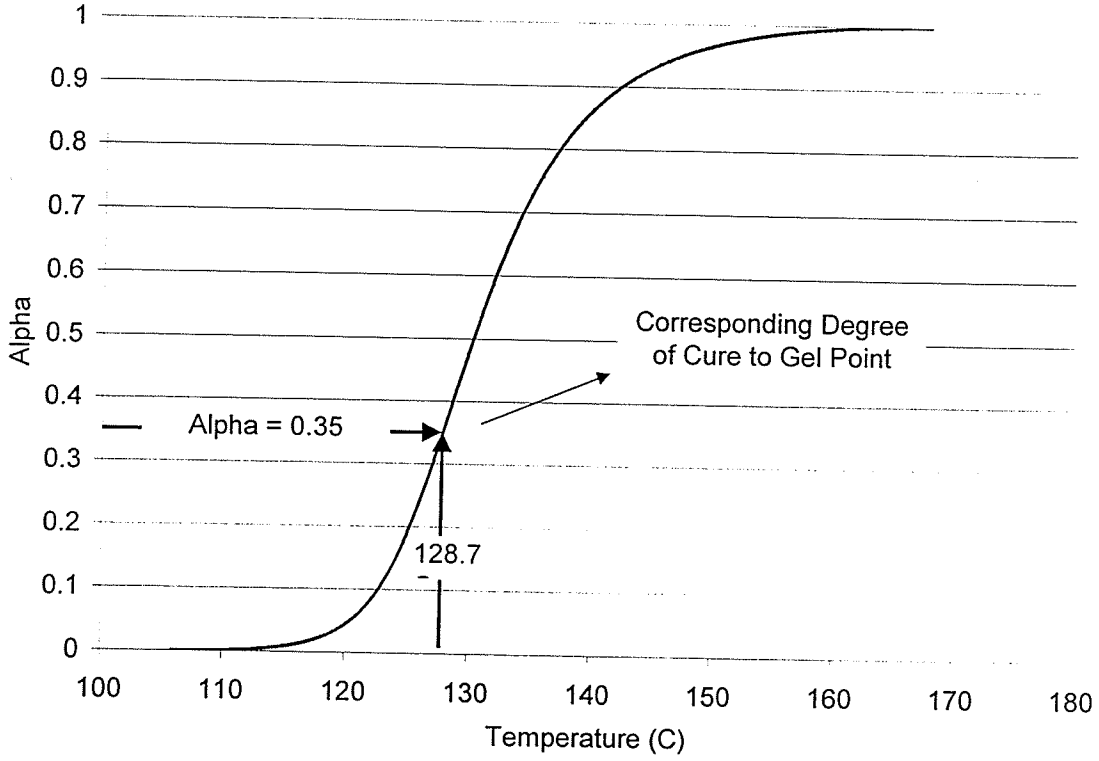

Figure 5.10: Alpha versus Temperature for a Dynamic Scan of 2°C/min. (Showing that gelation occurs at alpha =0.35)

Classification	Properties	Measured Values
Physical (from manufacturer data sheets)	Density Fiber Volume Fraction	Resin: 759.1, Fiber: 1760 57 % (unidirectional) 50% (woven)
Cure Kinetics (model from Equation 5.1)	Total heat of reaction (kJ/kg ^{o}C) Model constants for Equation 5.1	Hr = 128 k = Z exp (-E/RT) Z = 1.502E+08 sec^{-1} E = 78.809 kJ/mol m = 0.63722 n = A exp (B*T) A = 0.0013366 B = 0.0175435 K^{-1} Alphafinal = C exp (D*T) C = 0.0011557 D = 0.017402 K^{-1}
Thermo-Physical	Specific heat capacity (J/kg ^{o}C) Cure Shrinkage CTE of Composite x 10-6 / ^{o}C Resin Degradation Temperature	Cp = 0.6589 + 0.0025*T csc22 = csc33 = -18.0635 α^5 – 40.6545 α^4 + 25.026 α^3 – 2.50645 α^2 – 2.6813α CTE = 1.677E-8*T^2 – 1.314E-6*T +9.439E-5 250 ^{o}C

Rheological	Viscosity (From Equation 5.6)	A_μ = 6.5452E-12 Pas E_μ = 90477.8 J K/mol α_{Gel} = 0.533 A = 3.185 B = 2.359
	Gel Point	α = 0.35

Table 5.1: Summary of Material Properties for F155 Resin with Toho T300 Fibers

model. The process model was run for Cycles 1 and 2, as shown in Figure 4.9 and 4.10, with pressures of 45 and 85 psi. Figures 5.11 and Figure 5.12, show plots of experimental and predicted temperature data for Cycle 1 at 45 psi and Cycle 2 at 85 psi respectively.

Figure 5.11 and 5.12 show the cure cycle used to model the actual autoclave air temperature. The process model cycle does not completely capture the slight overshoot that occurs as the autoclave reaches the hold temperature. Regardless, the model predicts the lag in part temperature during heating and cooling. This proves that the heat transfer boundary conditions were modeled correctly. The maximum error between experimental and model data was only 10°F (5.5°C). This error occurred due to the imperfect modeling of the actual cure cycle but is not enough to adversely affect the predictions made by the thermochemical module.

5.2.2 Optimization Results

This section provides the results from the series of optimization runs that are described in Section 4.4.4 and summarized in Table 4.7. A summary of results for all optimization runs can be found in Appendix B. As mentioned previously, all runs have been done with the temperature, degree of cure and time fitness functions turned on. Temperature and cure gradient functions were turned off based upon initial results as summarized in Table B.3. From this table, the maximum temperature gradient for any run was 8.89°F (5°C) while the maximum cure gradient was 0.02. This temperature

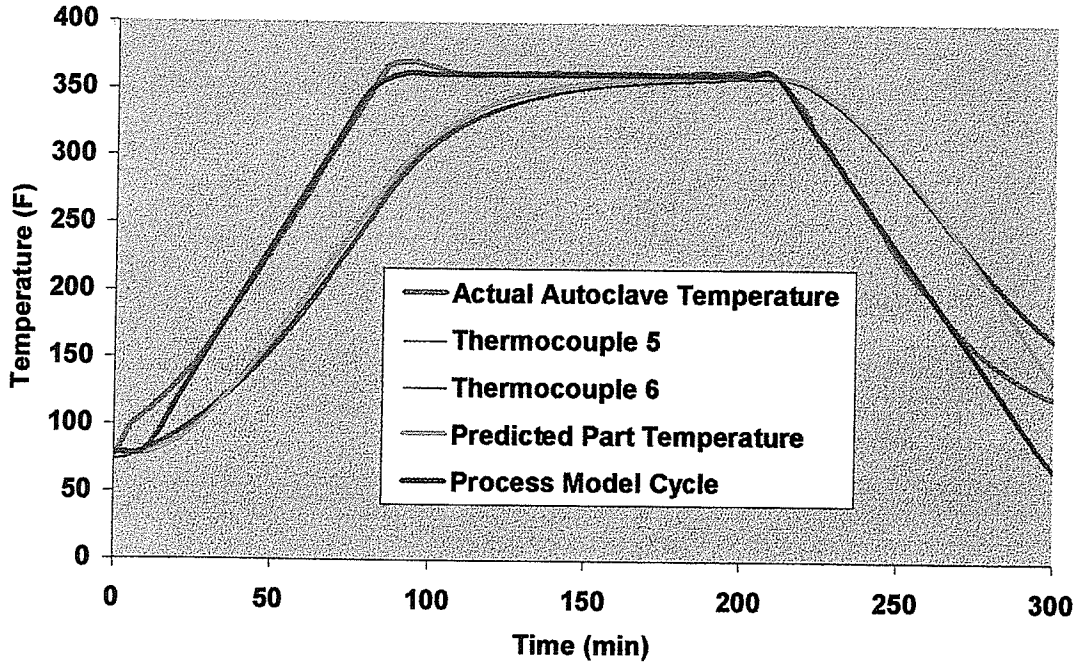

Figure 5.11: Plot of Actual and Predicted Part Temperature versus Time for "Tool C" Using Cycle 1 (from Figure 4.9) at 45 psi

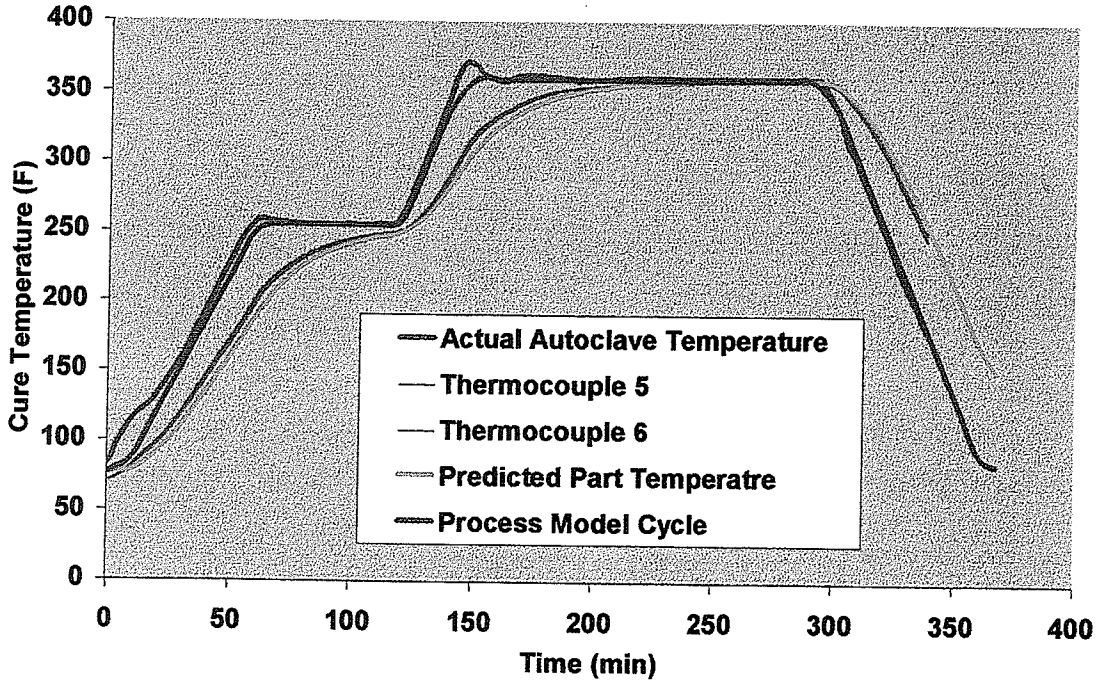

Figure 5.12: Plot of Actual and Predicted Part Temperature versus Time for "Tool C" Using Cycle 2 (from Figure 4.10) at 85 psi

gradient is less than those used by Rai and Pitchumani (15) who deemed acceptable temperature gradients to be in the range of 10°C to 20°C for various optimization runs.

Since each of the fitness functions (cure, temperature, and time) has a maximum value of 1, the maximum possible total fitness is 3. It should be noted that no checks are performed to ensure that parameters meet the minimum requirements although all runs after Run 4 meet the minimum requirements. Two important values are referred to in the following sections - total number of populations and total number of cycles (iterations). The total number of populations, as shown in Equation 5.7, is the product of the number of generations (NGEN) and the number of micro-cycles (NMGA).

$$\text{Number of populations} = \text{NGEN} * \text{NMGA} \tag{5.7}$$

Where: NGEN – Number of Generations

NMGA – Number of micro-cycles

NGEN and NMGA values are inputs in GAINPT.dat. Table 4.7 summarizes the settings for these parameters for each optimization run. The total number of populations is the value shown on the x-axis of the fitness graphs that are presented in the following sections. All optimization runs used populations that each had 10 members, therefore the total number of cycles (iterations) in a given optimization run is:

$$\#\text{ of cycles (iterations)} = \text{NPOP} * \text{NGEN} * \text{NMGA} \tag{5.8}$$

Where: NPOP – number of members in a population (set to 10 for all runs)

NGEN – Number of Generations

NMGA – Number of micro-cycles

It should also be mentioned that all graphs of the optimal cure cycles are for the air temperature in the autoclave and include the manufacturer cycle as a reference point.

5.2.2.1 Convergence

Initial runs of the optimizer were performed to determine the number of iterations to be used for subsequent runs. Tse and Chan (20), have shown that populations sizes of 10 member were optimal. Therefore, all optimization runs were performed with populations with 10 members (NPOP).

Figure 5.13 shows the fitness for runs 1, 3 and 4. Runs 1 and 3 had the same settings (Regular GA) except Run 1 was for 2000 populations and Run 3 stopped at 400 populations. As shown in Figure 5.13, fitness for Run 1 increased until approximately 800 populations. However, the fitness only increased by 0.04 between 400 populations and 800 populations. The extra 400 populations typically translated into 24 hours of time on a Pentium III 933MHz computer. Therefore, the initial decision was to run all remaining runs at 400 populations. Before this decision was finalized, Run 4 was used to verify that the micro-GA mode would also converge in the same number of populations. As Figure 5.13 shows, both the regular GA and micro-GA modes converged around 400 populations. Therefore, the decision was finalized that 400 populations would be used for the remainder of optimization runs unless another run proved otherwise.

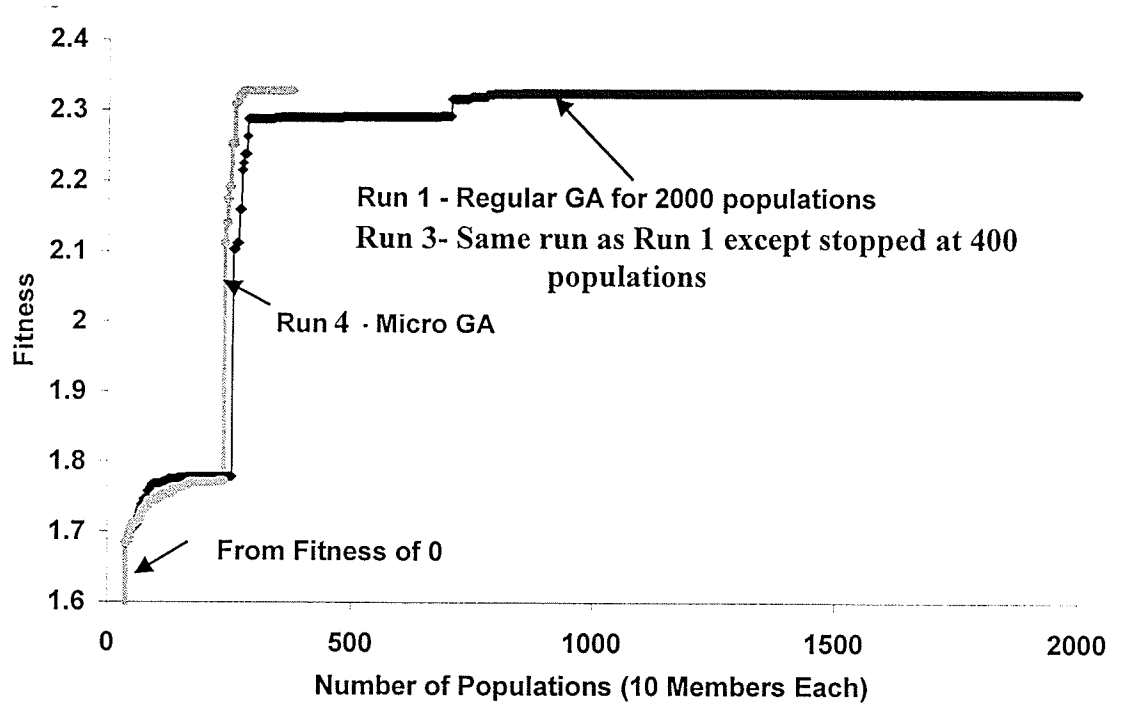

Figure 5.13: Plot of Fitness vs. Number of Populations for Runs 1,3 and 4

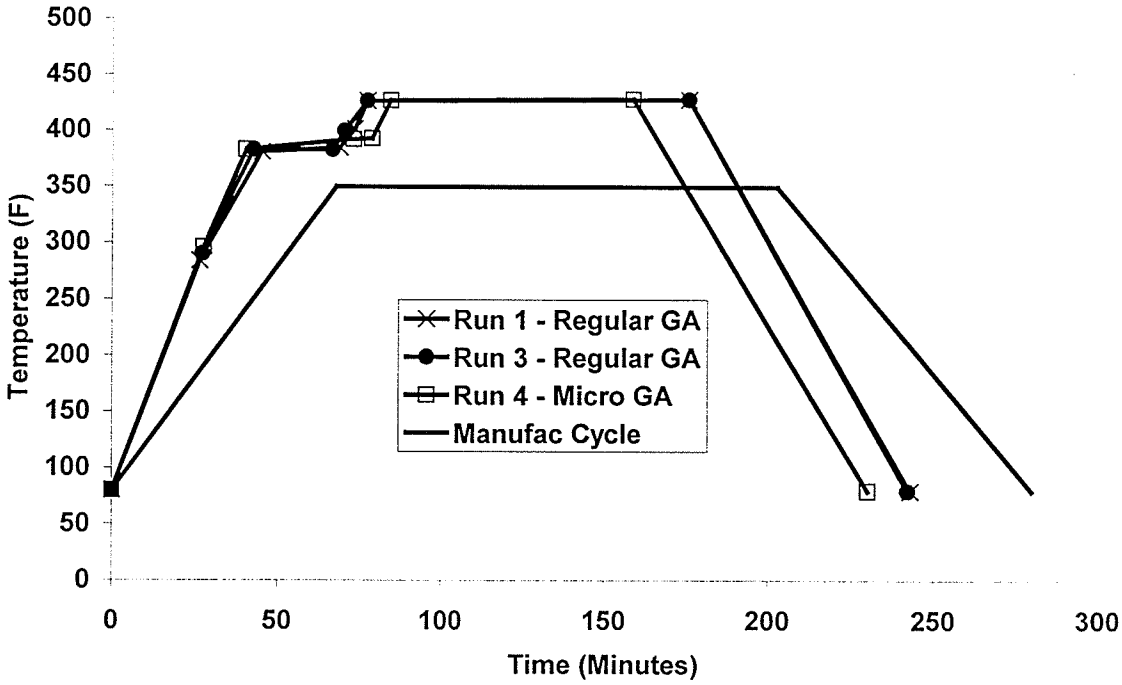

Figure 5.14: Optimal Cycles for Runs 1-4 versus Manufacturer Cycle

Figure 5.14 shows the optimal cycles generated by Runs 1, 3 and 4 as compared to the manufacturer's recommended cycle. Optimal cycles for Runs 1 and 3 are almost identical because the settings are essentially the same. The important quality to note about the cycles is that the maximum fitness temperature of 370°F (TempMax in QAMeas.txt – Figure 4.4) was exceeded. However, the cycles does not exceed the maximum autoclave air temperature of 430°F (Figure 4.5). The difference between the maximum autoclave air temperature and the maximum fitness temperature (part temperature) is that the GA uses the maximum curing temperature to set an absolute temperature that any given cycle cannot exceed. If the cycle exceeds this temperature, then the cycle is assigned a low fitness value and the cycle is therefore not run through the process model. The maximum fitness temperature is used by the fitness function to calculate the temperature fitness function for a given cycle after a cycle passes through the absolute limit set by the maximum curing temperature.

There were 2 reasons for setting the maximum autoclave temperature above the maximum fitness temperature. The first reason was that the autoclave air temperature should be able to exceed the maximum part temperature in order to speed up a cure as long as the part does not exceed the degradation temperature. The second reason stems from the problem that was discussed in Section 4.4.4.4 with regards to the maximum heating rate. The autoclave air temperature limit was increased in the GAINPT.dat cycle so as to increase the probability that the GA could generate a random cycle that passed this absolute limit. It was speculated that these settings would permit the GA to pass more cycles through the absolute limit of 430°F at which point the GA would optimize

towards cycles that were below the maximum part temperature of 370°F. Unfortunately, this was not the case and it was decided that injecting cycles could help alleviate this problem. The effects of input cycles on the optimizer is discussed further in Section 5.2.2.3.

Another interesting feature shown in Figure 5.13 is the sharp jump in fitness that occurred at approximately 250 populations. This sharp jump was caused by one of the parameters (cure, time or temperature) moving from the unacceptable region of the fitness function to the acceptable region of the function. The exponential functions within the unacceptable region were designed to the cause a sharp increase as a parameter moves towards the acceptable region.

The final feature to note in Figure 5.13 is that all the optimal cycles have a hold in them. This is a unique feature that has not been seen in many of the other optimization work performed by others. Many other optimizers generate functions that have the maximum ramp rate followed by an immediate cool down ramp.

5.2.2.2 GA versus Micro-GA

Tse and Chan (20) have shown that micro-GAs can converge to a higher fitness than regular GAs. As setting were changed to study various effects in the optimizer, runs that were set to operate in regular GA mode (Runs 3, 5, 7, 9 and 11) were later set to operate using the micro-GA mode to compare the effects (Runs 4, 6, 8, 10 and 12). Figures 5.15 to 5.18 show the fitness plots for the following pairs of regular GA and

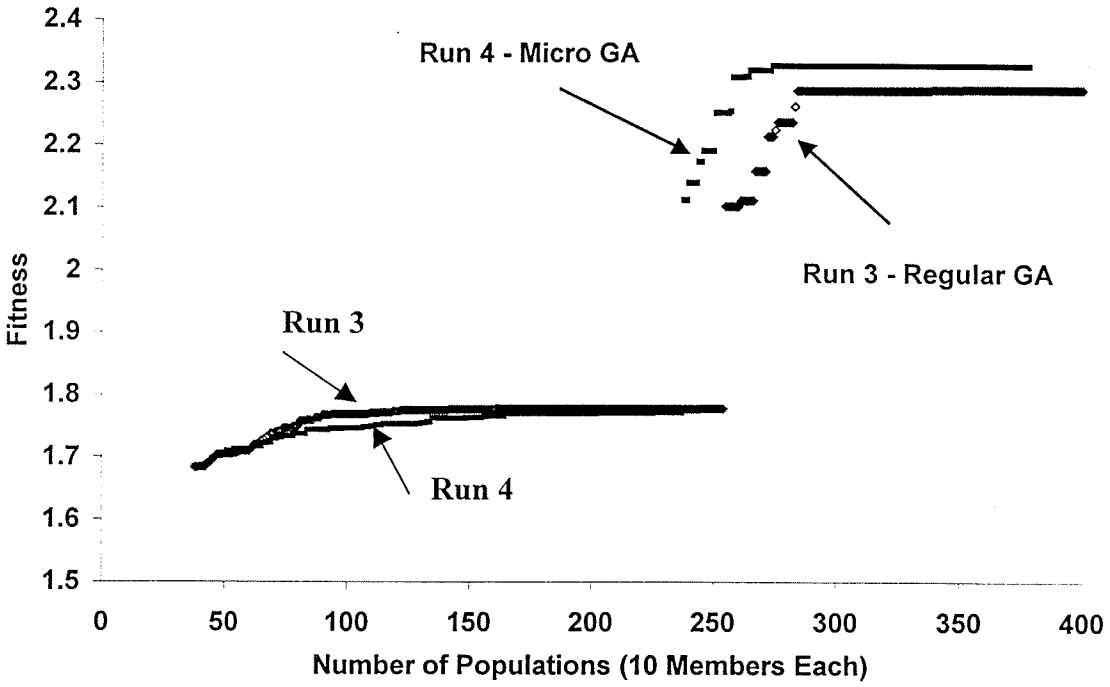

Figure 5.15: Plot of Fitness versus Number of Populations for Runs 3 and 4

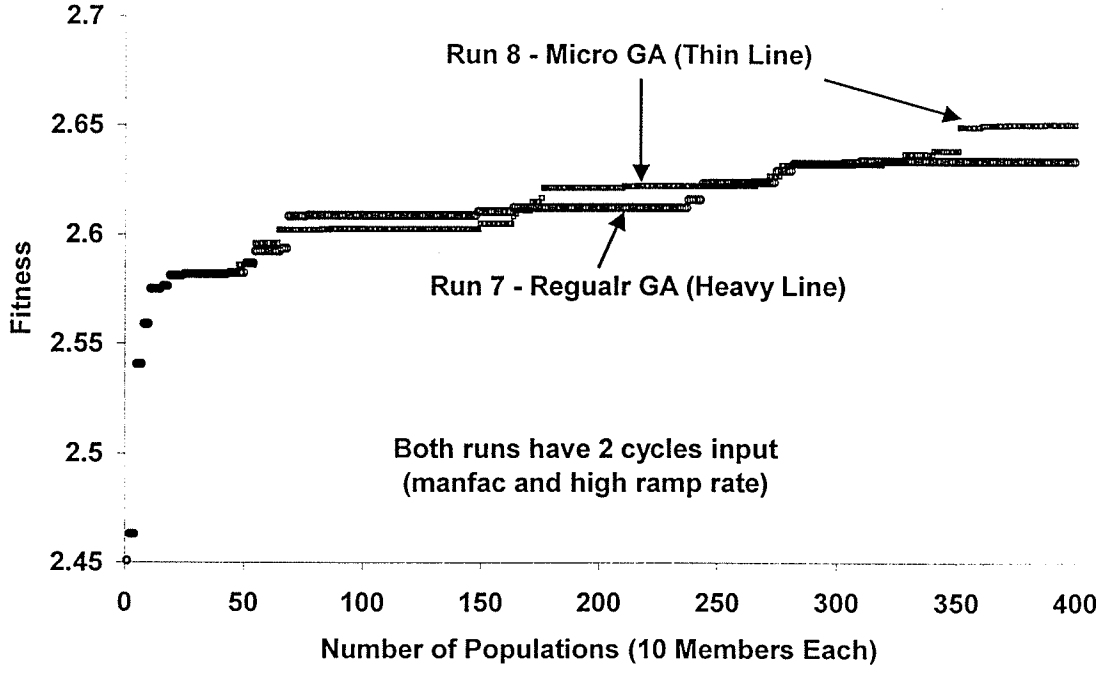

Figure 5.16: Plot of Fitness versus Number of Populations for Runs 7 and 8

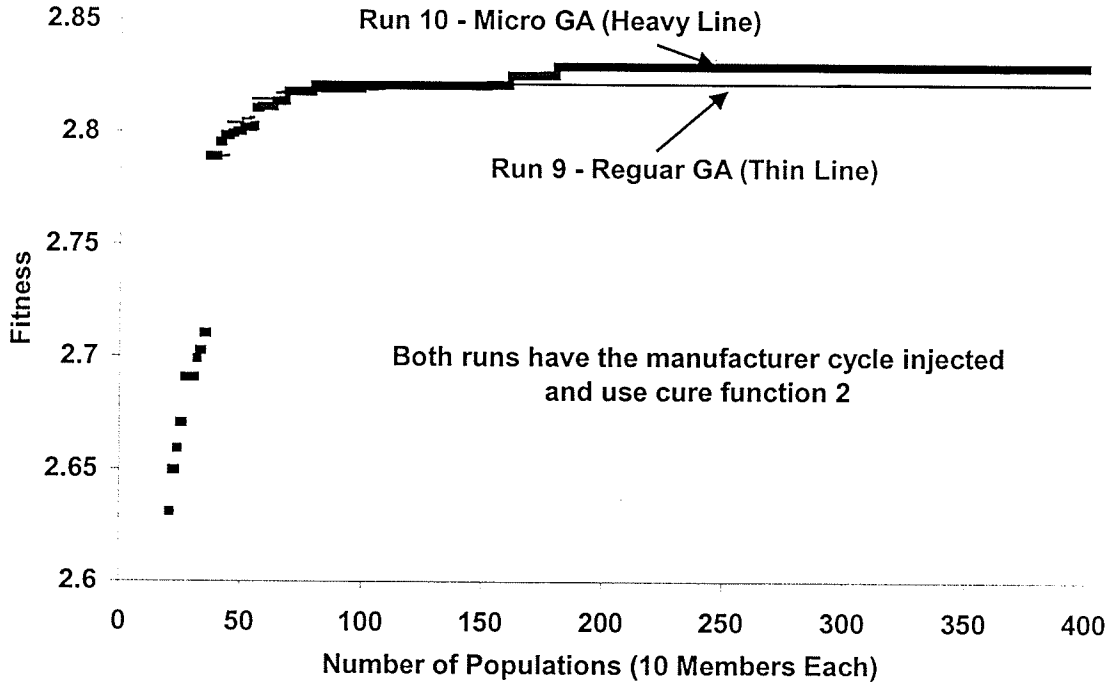

Figure 5.17: Plot of Fitness versus Number of Populations for Runs 9 and 10

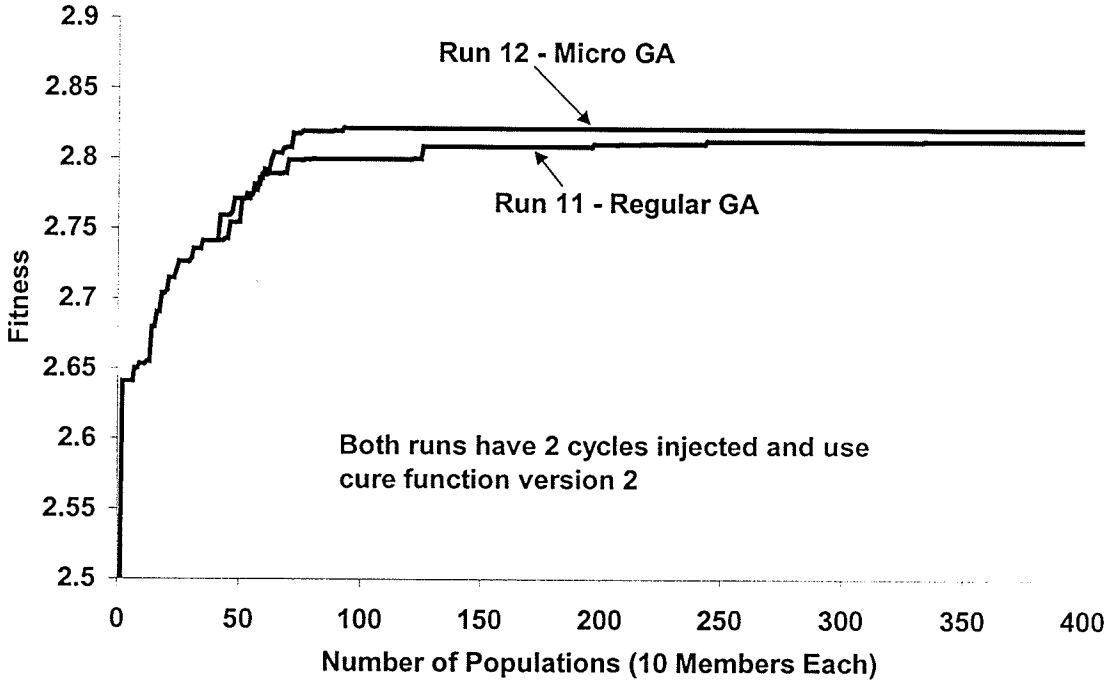

Figure 5.18: Plot of Fitness versus Number of Populations for Runs 11 and 12

micro-GA runs (Run 3-4, Run 7-8, Run 9-10 and Run 11-12). Results from Runs 5 and 6 were not plotted as the curves were too close to distinguish although the optimizer output files show that the micro-GA provides a slightly higher fitness value.

Figures 5.15 to 5.18 show that in each case, the fitness of the optimal cycle generated in micro-GA mode is moderately better than the regular GA result. This result is also evident from Table 5.2 which summarizes the optimal cycle times for each pair of runs. Each pair of runs shows that the cycle times are shorter for micro-GAs when compared to the regular GAs. An improvement of 19 minutes is seen in the pairing of runs 7 and 8. The better performance of the micro-GA versus a regular GA lead to the decision to perform all subsequent optimizations in micro-GA mode after Run 11. As well, the results presented in subsequent sections of this chapter will only show the micro-GA result of the pairs listed in Table 5.2

5.2.2.3 Influence of Input Cycle

Figure 5.19 shows the optimized cycles for Runs 4, 6, and 8 while Table 5.3 shows the optimization data from each run. Each cycle had the same settings (micro-GA) except Run 4 had no injected cycle, Run 6 had the manufacturer cycle injected and Run 8 had both the manufacturer and high ramp rate cycles injected. Figure 5.19 illustrates that optimization runs that had injected cycles (Run 6 and 8), all had times shorter than the manufacturer cycle achieved because of time savings during ramp up. Run 4, without an

Run	Description (Settings are the same for a given pair)	Optimal Cycle Time (min)
Manufacturer Cycle		280
3 (Regular GA) 4 (Micro GA)	Baseline settings – no injected cycles or changes in fitness functions	242* 230*
5 (Reg GA) 6 (Micro GA)	Manufacturer cycle injected	279 277
7 (Reg GA) 8 (Micro GA)	2 cycles injected	284 265
9 (Reg GA) 10 (Micro GA)	Manufacturer cycle injected and cure function changed to version 2	265 258
11 (Reg GA) 12 (Micro GA)	2 cycles injected and cure function changed to version 2	244 236

*- denotes cycle that caused resin to degrade

Table 5.2: Optimal Cycle Times for Runs 5 to 12

Run	Total Time (min)	T_{max} (°F)	α_{min}	Fitness Temp	Fitness Alpha	Fitness Time	Total Fitness
4	230	NA	NA	NA	NA	NA	NA
6	277	370	0.960	1	0.876	0.782	2.658
8	265	370	0.958	1	0.069	0.782	2.65

Table 5.3: Optimization Data for Runs 4,6 and 8

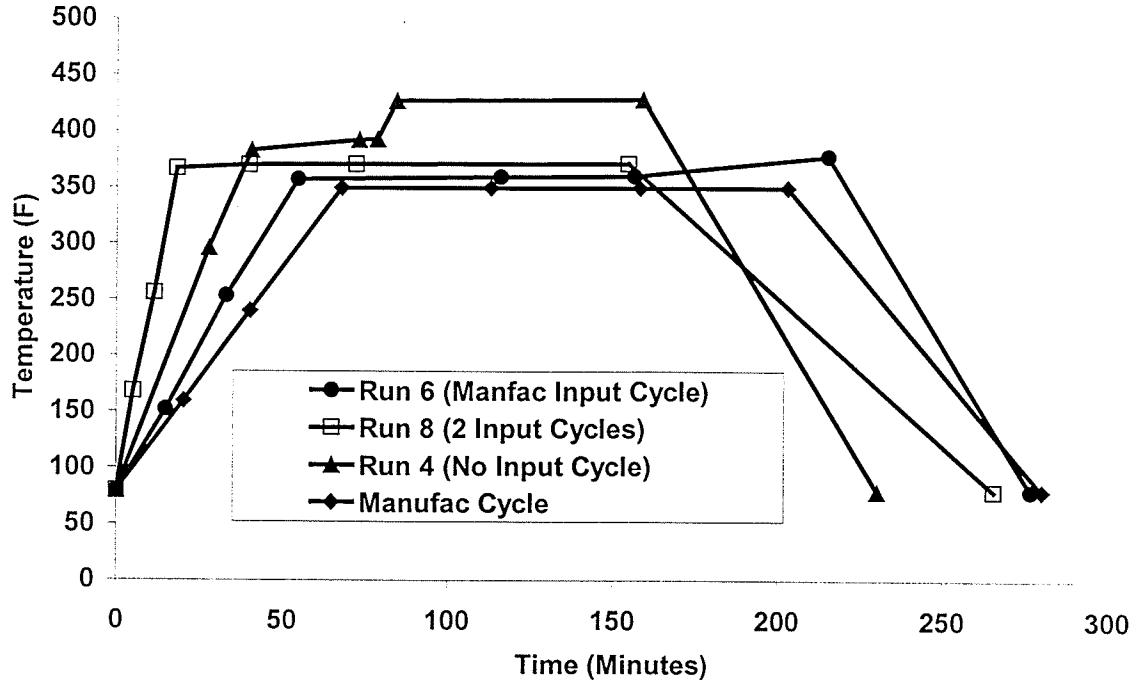

Figure 5.19: Optimal Cycles With and Without Injected Cycles

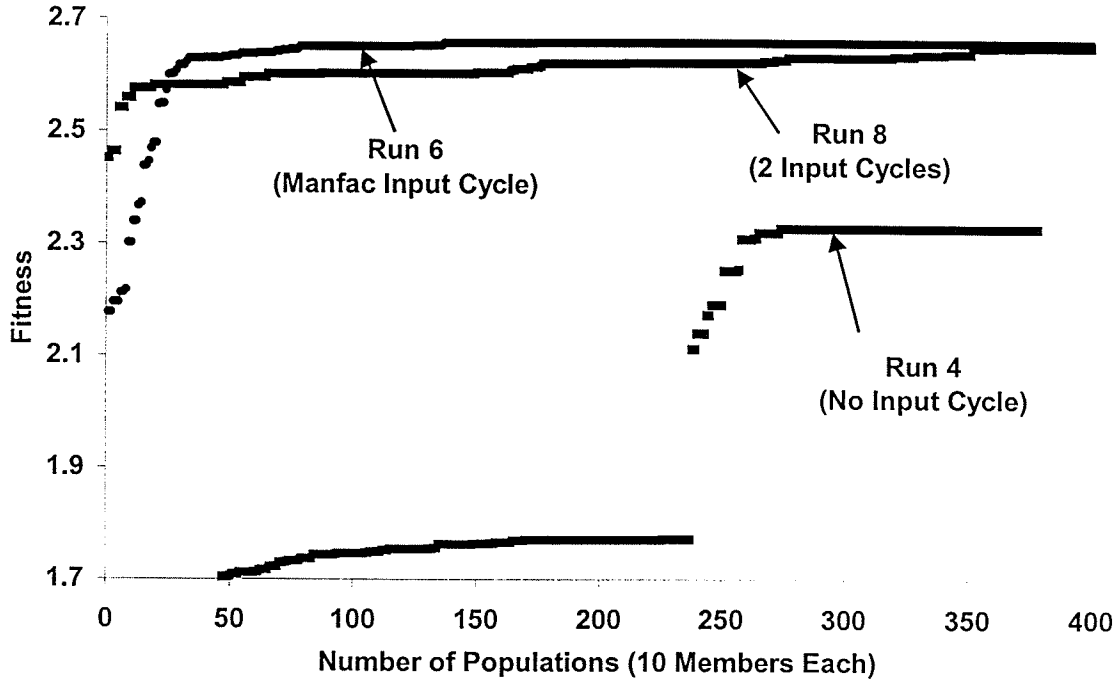

Figure 5.20: Fitness Plot for Runs 4, 6 and 8

injected cycle, was shorter than the manufacturer cycle but was hot enough to cause the resin to degrade.

Another interesting feature in Figure 5.19, is that the air temperature cycle for Run 6 (manufacture cycle injected) slightly exceeds the maximum part temperature. Although, the air temperature exceeds the maximum part temperature, Table 5.3 shows that the part temperature only reaches the maximum temperature of 370°F and therefore the cycle is acceptable. As mentioned in Section 5.2.2.1, by setting the maximum autoclave air temperature higher than that of the maximum part temperature, it was speculated that air temperature cycles above the maximum part temperature could be generated without causing the resin to degrade.

An advantage of the injection of the high ramp rate cycle is that the heating rate could be increased to 18°F/min from 7.9°F/min. Previous optimization runs were limited to heating rates of 7.9°F/min due to the problem discussed in Section 4.4.4. Even optimization runs with the manufacturer cycle were limited to ramp rates of 7.9°F/min because the heating rate for that particular cycle was 4°F/min. Such low ramp rates do not exploit the capabilities of the autoclave thereby reducing the potential to optimize a cycle. One area in which time savings can be made is in the cool down segment. Notice in Figure 5.19 that the cool down segment is not very steep. This is because the optimizer is trying achieve a degree of cure of exactly 0.92. The optimizer does this by completing the cure as the cycle begins to cool down. One possible way to create a faster

cooling would be to add a segment in the cool down stage. Regardless, there is still a time savings when injecting the high ramp rate.

Figure 5.20 shows the fitness plot for Runs 4, 6, 8. From this figure, it can be seen that optimization runs with input cycles experience a sharp increase in fitness much sooner and to a higher level of fitness than optimization runs without cycle injection. The earlier increase in fitness values was achieved because the probability of generating good cycles in future generations, via cross-over and mutation, increases substantially when good cycles are injected in the first population. If the optimizer is allowed to start at random, then the chances of finding a good cycle is random and so convergence requires more populations.

5.2.2.4 Effect of Inputs in GAINPT.dat File

Several options can be changed in the GAINPT.dat file. This section will provide the results from optimizations in which the random seed, PMUT (probability of mutation) and NMGA (number of micro generations) were changed.

The random seed value in GAINPT.dat is the starting point for the random number generator. By changing this value, the place in which the random number generator starts changes with it. Run 2 was performed to study the effect. It was found that the fitness development for Run 2 and Run 3 (identical settings except random seed value) were nearly identical and so are not plotted together. However, fitness plots and

optimal cycles are shown in Figures B.5 to Figure B.8. Tse and Chan (20) found that after 100 trials with different seeds that the results were all very similar. For this reason, the random seed number was not changed for all optimization runs after Run 2.

The next GAINPT.dat variables that was looked at were PMUT (probability of mutation) and NMGA (number of micro-cycles). Figures 5.21 and 5.22 show the fitness plots and optimal cycles generated for runs 12, 16 and 17. Runs 16 and 17 used the same settings as Run 12 except PMUT was changed to 0.1 in Run 16 and NMGA was changed from 10 to 40 in Run 17. NGEN was adjusted to 10 from 40 so that the run had the same number of populations (400) as all the other runs. Figure 5.21 illustrates that PMUT slows the convergence rate of the optimizer as the probability for mutation is decreased which reduces the amount of randomness into the optimizer. Unfortunately, Run 16 was cut short but it is evident that if the trend continues, the optimizer will not be able to achieve the fitness levels as seen in Run 12 and 17.

Figure 5.22 shows that by increasing the number of micro-cycles (NMGA), as was done with Run 17, the optimal cycle is much shorter than the similar setup in Run 12. Run 17 provides an optimal cycle time of 203 minutes which is the second best and 33 minutes shorter than Run 12. This decrease in time can be attributed to the added randomness provided by the increased number of randomly generated cycles from the micro-GA. As discussed in Section 3.1.1, everytime a micro-cycle starts, the micro-GA

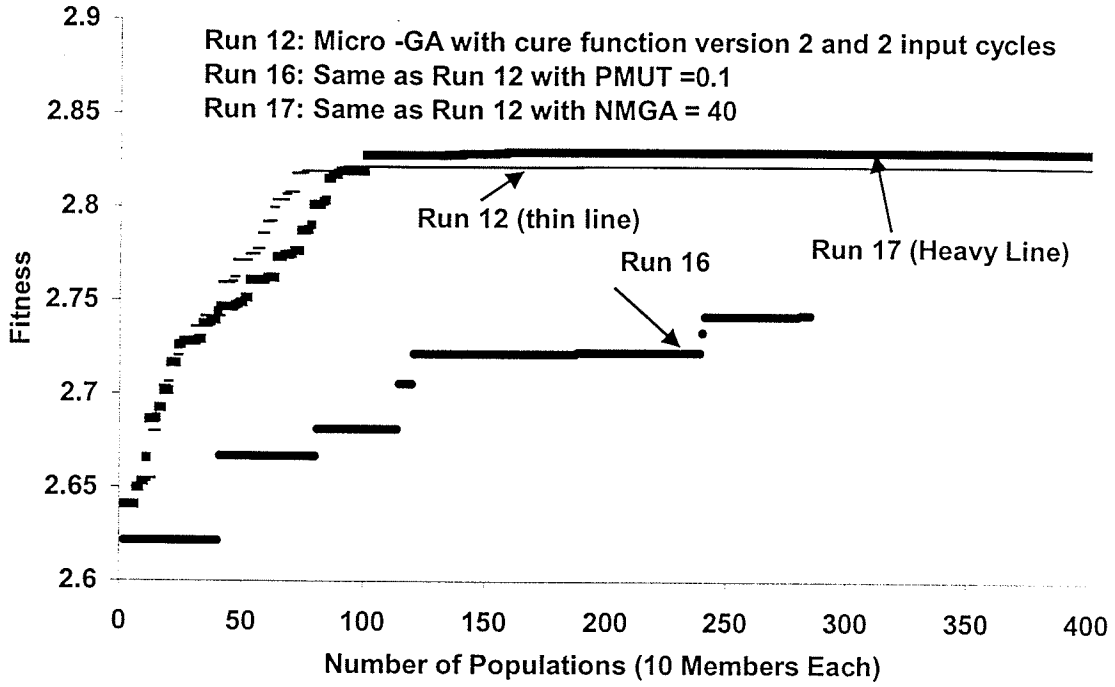

Figure 5.21: Fitness Plot for Runs 12, 16 and 17

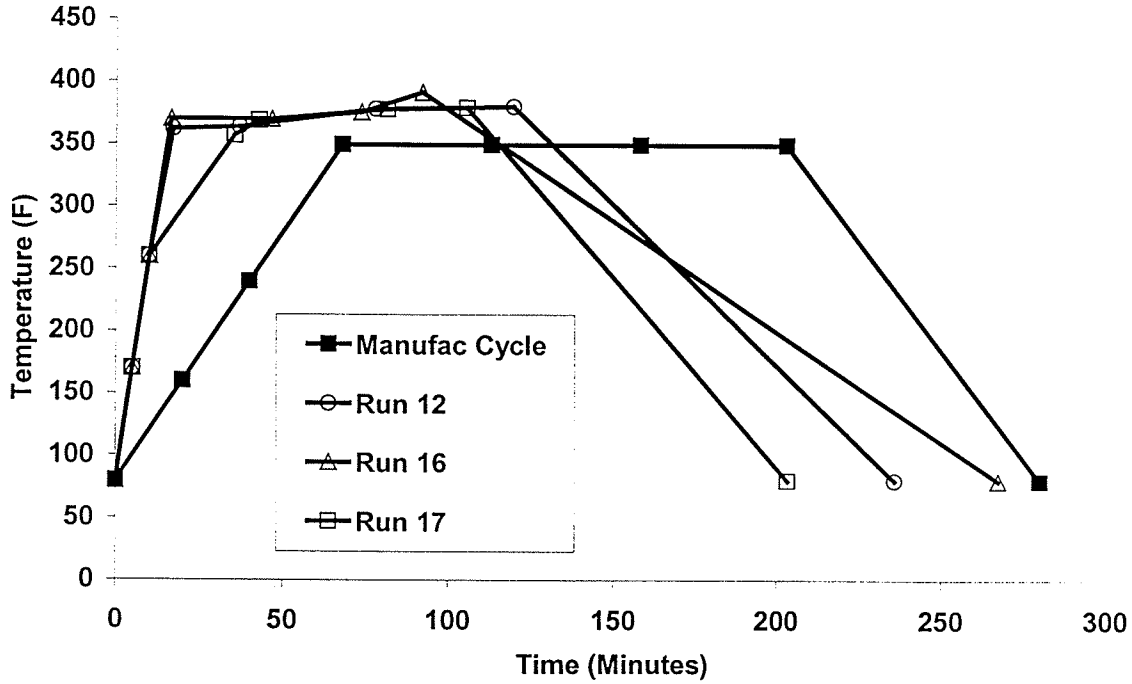

Figure 5.22: Optimal Cycles for Runs 12, 16 and 17

develops a new random set of cycles. These cycles help to inject new possibilities into an optimization run that has "stalled". Figure 5.21 shows Run 12 as stalled while Run 17 continues to increase in fitness.

5.2.2.5 Influence of the Shape of Fitness Functions

The fitness functions in the fitness equation have an important role in leading the GA towards the optimal solution. This section describes the effects of changing fitness functions individually and in combination. The various versions of the cure and time fitness functions that are referred to in this section are described in Chapter 4.

Figure 5.23 and 5.24 shows the fitness plots and optimal cycles for Runs 8, 12, and 13. These runs have the same setup (micro-GA with 2 input cycles) with the exception that Run 8 has cure function version 1, Run 12 has cure function version 2, and Run 13 has cure function version 3. These figures illustrates that version 2 of the cure function gives the shortest cure cycle as well as the highest fitness. The cycle time for version 2 is 19 minutes shorter as compared to version 1 while version 3 provides only a 1 minute savings.

Cure function version 1 resulted in the longest cycle time because the optimizer was trying to optimize towards higher degrees of cure as seen in Table 5.4. This was because the rate of fitness increase for cycles with higher degrees of cure was faster than the decrease in fitness from the time function due to the resulting longer cures. In other

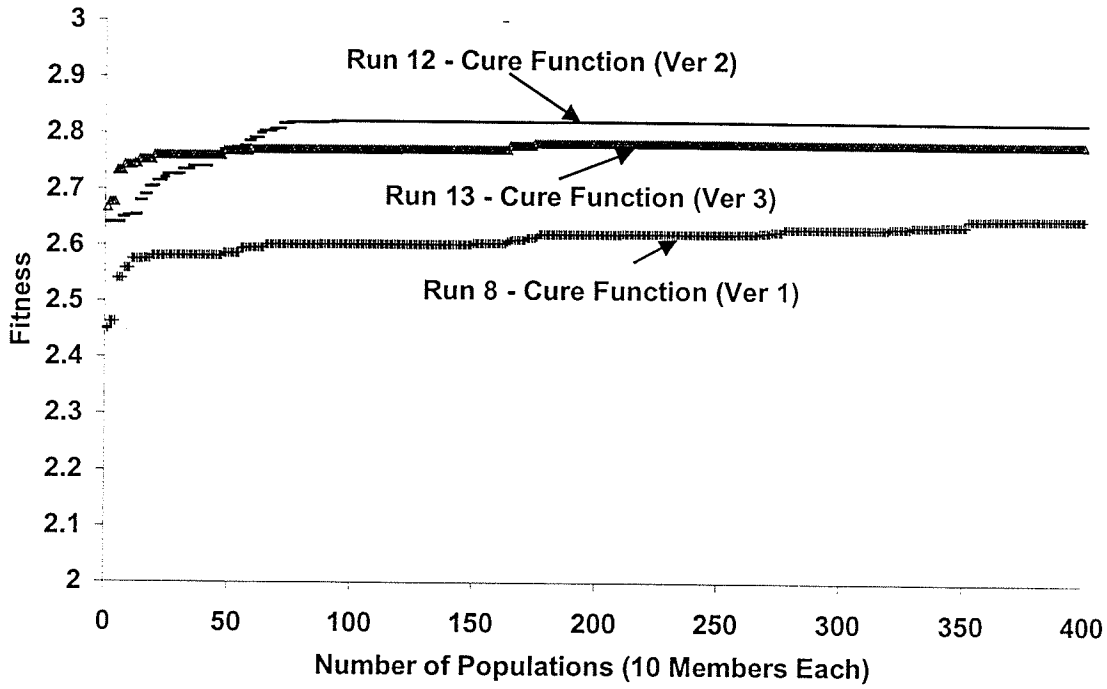

Figure 5.23: Fitness versus Number of Populations for Run 8, 12 and 13

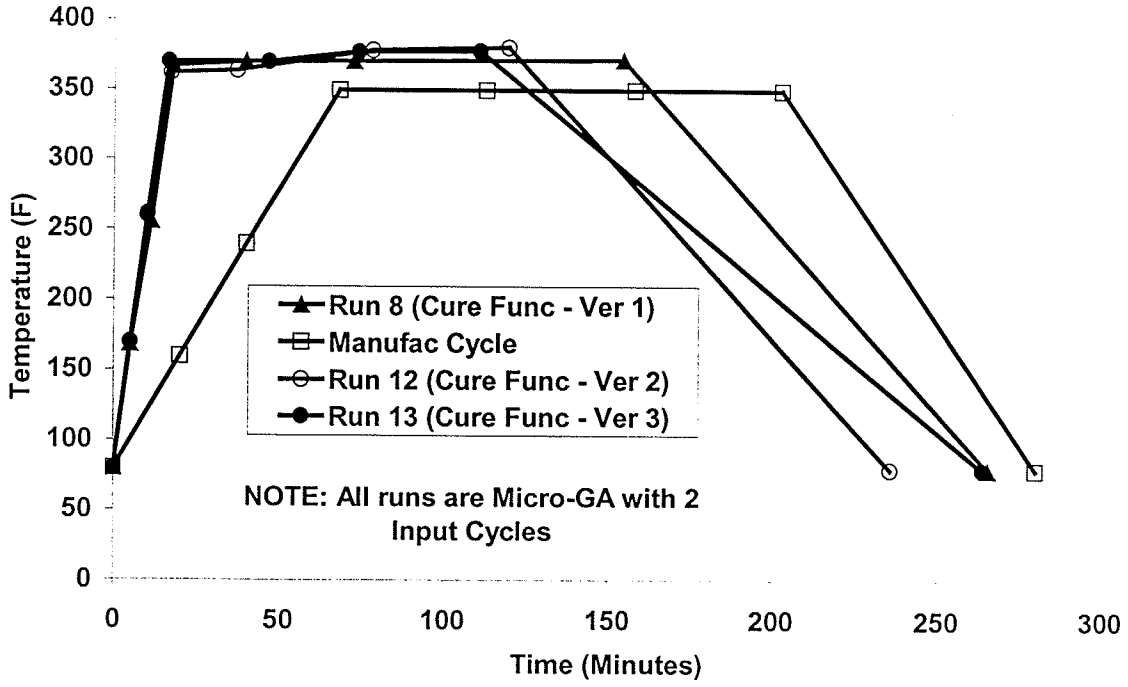

Figure 5.24: Optimal Cure Cycles for Run 8, 12 and 13

Run	Total Time (min)	T_{max} (°F)	α_{min}	Fitness Temp	Fitness Alpha	Fitness Time	Total Fitness
8	265	370	0.958	1	0.069	0.782	2.65
12	236	369.97	0.92	1	1	0.822	2.82
13	264	369.64	0.93	0.994	1	0.790	2.78

Table 5.4: Optimization Data for Runs 8, 12 and 13

words, for optimization runs with cure function version 1, a cure cycle that had a minimum degree of cure 0.92 received a cure fitness of 0.75 while a degree of cure of 1 received a cure fitness of 1. Therefore, fitness increases by 0.25 when the degree of cure rises from 0.92 to 1 whereas the time function increases by 0.25 in 310 minutes. This difference in the rate of fitness change is the reason for the long length of the cycle provided by cure function version 1.

Cure function version 2 shortens the cure cycle by preventing the optimizer from optimizing to a higher degree of cure than 0.92. Since degree of cure is time-dependant, given the temperature is the same, lower degrees of cure will require shorter cure cycles. As shown in Table 5.4, cure function version 3 provides similar results as cure cycle version 1 in that the optimizer produces a cure cycle where the degree of cure is higher than 0.92. This naturally makes the cure cycle longer.

The next function studied was the time function. As was mentioned above, there appeared to be a deficiency in the rate of increase in fitness for the time function. A second version of the time function was created to study this problem. Figure 5.25 and 5.26 show the fitness plot and optimal cycles for Runs 12 to 15. All of these runs were setup in micro-GA mode using time function version 2 and 2 input cure cycles. The difference between runs was that Run 12 and 14 use version 2 of the cure function and Runs 13 and 15 use version 3 of the cure function. As shown in Figures 5.25 and 5.26, optimization runs with time function version 2 (Runs 14 and 15) have higher fitnesses and shorter cure cycles as compared to the corresponding optimization runs with time

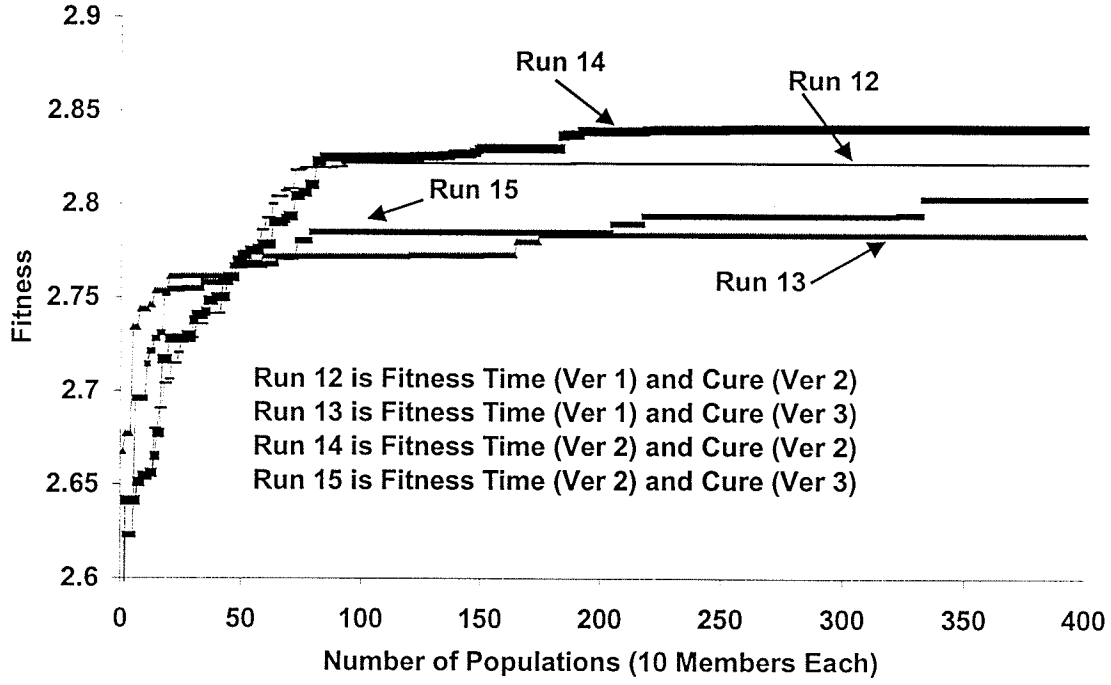

Figure 5.25: Fitness versus Number of Populations for Run 12-15

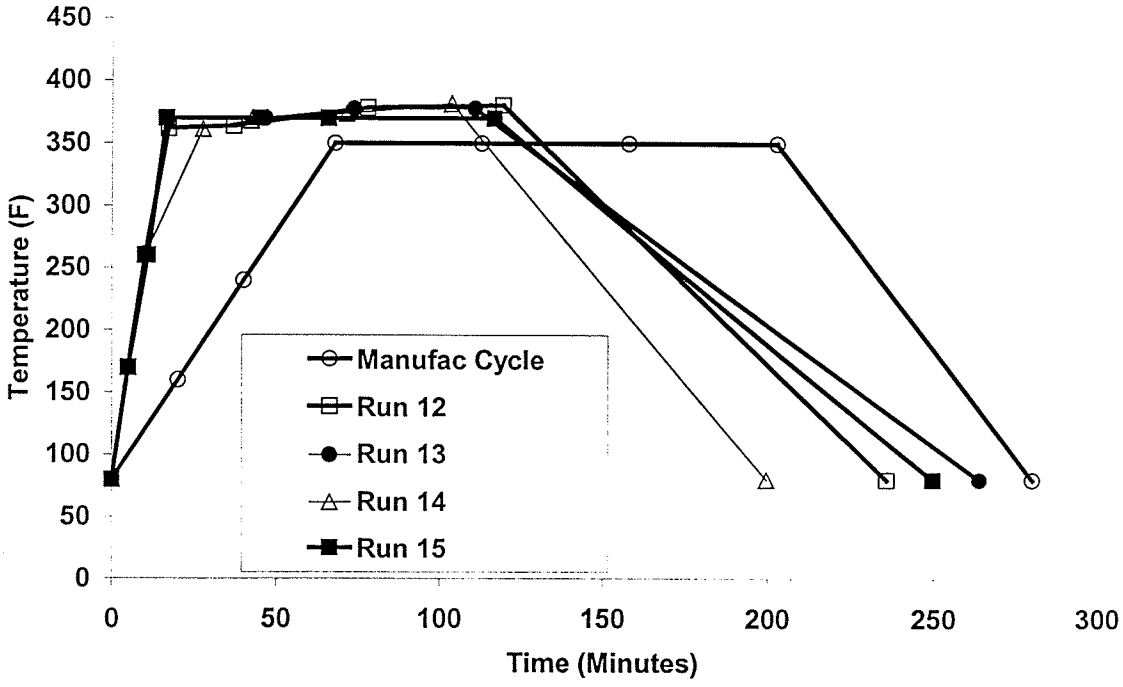

Figure 5.26: Optimal Cure Cycles for Runs 12-15

function version 1 (Runs 12 and 13). This suggests that the slope of the line within the acceptable range of the time function is important.

5.2.2.6 Summary

It was found that the optimizer ran the best in the micro-GA mode. This mode injected random cycles throughout the run which aided in solutions that had moderately higher fitnesses and shorter cure cycle times. The micro-GA did not seem to have an effect on the number of populations required for convergence. Both the regular GA and micro-GA modes required 400 populations for convergence.

Some of the options found in GAINPT.dat were found to have an effect on the final solution. The option, PMUT (Probability of Mutation), had an effect on the rate at which the optimizer converged. A lower setting, as was used in Run 16, dramatically slowed the rate of convergence of the optimizer as well as lower the final fitness value. The GAINPT.dat option that was found to have the largest impact was the number of micro-cycles (NMGA). By changing this value from 10 to 40, the optimizer developed the second shortest cycle when used with cure function version 2 and the 2 input cycles. The increased number of micro-cycles increases the amount of random cycles generated during an optimization run. These random cycles appear to delay the "stalling" of the optimizer which results in better solutions. The final option, the random seed, appeared to have no bearing on the final solution and so was changed for only one run (Run 2)

Injected cycles have been shown to have an influence on shortening cycle times. Runs that injected both the manufacturer cycle and the high ramp rate cycle were shown to produce shorter cycles than runs that injected the manufacture cycle alone. The high ramp rate cycle was required to increase the maximum heating rate from 7.9°F/min to 18°F/min. This allowed for the optimizer to better utilize the potential of the autoclave thereby reducing optimal cure cycle times.

The shape of the fitness function was shown to help improve the results provided by the optimizer. Cure function version 2 shortened the optimal cure cycle times by driving the optimizer towards a cycle that produced the minimum required degree of cure of 0.92. Cure cycles that produce lower degrees of cure are naturally faster than cycles that produce higher degrees of cure given heating and cooling rates are approximately the same.

It was also shown that time function version 2 was very effective in shortening the optimal cure cycle times. Run 14 proved that time function version 2 when used with cure function 2 produces the best optimal cure cycle by producing time savings of 80 minutes. This example shows that the slope of the time function within the "acceptable" region has an effect on the final solution.

When the best components are brought together in Run 18, the shortest cycle time was found. Figure 5.27 shows that run 18 is 134 minutes which is less than half the manufacturer cycle (280 minutes) and 67 minutes shorter than the previous best (Run 14).

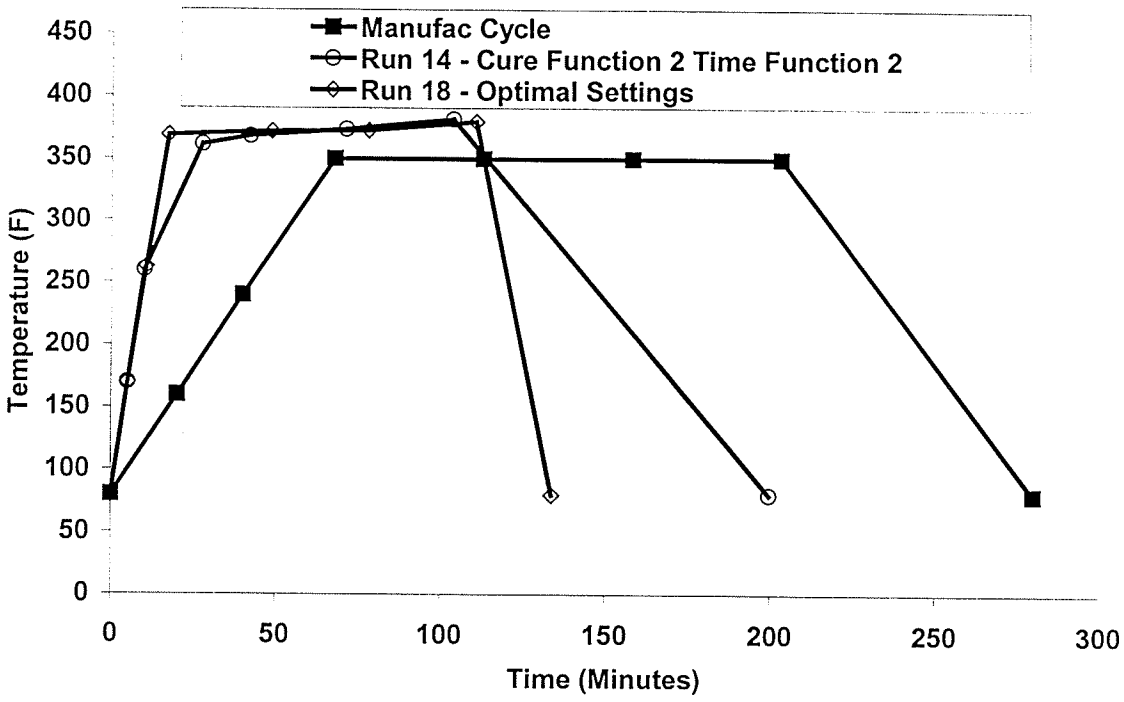

Figure 5.27: Optimal Cycles for Runs 14 and 18

CHAPTER 6

Conclusions

6.0 Introduction

This chapter outlines the conclusions drawn during this report. Thesis objectives as shown below have been achieved in full.

The primary objectives of this thesis can be itemized as:

a) develop the optimizer module for the APCS

b) Integrate the optimization module with the other modules of APCS

c) Demonstrate successfully the optimization capability of the optimizer and APCS

A secondary objective is:

d) Material characterization for Hexcel F155 resin reinforced with Toho T300 carbon fiber for its physical, thermal, rheological and mechanical properties.

6.1 Material Data

The material characterization of the material data for Hexcel F155 resin with Toho T300 carbon fiber was completed for the following list of properties. A summary of this can be found in Table 6.1.

- Cure Kinetics parameters
- Resin Degradation Temperature (T_{deg})
- Cure Shrinkage
- Coefficient of Thermal Expansion (CTE)
- Specific Heat Capacity
- Viscosity
- Gel point

The above list of material properties provides enough information to run the thermochemical module of the process model but not enough to run the flow and stress modules. When sufficient models are developed for flow and stress, the following is a list of parameters that would need to be characterized:

- Composite modulus versus degree of cure (E11, E22, and G12)
- Cure shrinkage in the fiber direction
- CTE in the fiber direction

Classification	Properties	Measured Values
Physical (from manufacturer data sheets)	Density Fiber Volume Fraction	Resin: 759.1, Fiber: 1760 57 % (unidirectional) 50% (woven)
Cure Kinetics (model from Equation 5.1)	Total heat of reaction (kJ/kg oC) Model constants for Equation 5.1	Hr = 128 $k = Z \exp(-E/RT)$ $Z = 1.502E+08\ sec^{-1}$ $E = 78.809\ kJ/mol$ $m = 0.63722$ $n = A \exp(B*T)$ $A = 0.0013366$ $B = 0.0175435\ K^{-1}$ $Alphafinal = C \exp(D*T)$ $C = 0.0011557$ $D = 0.017402\ K^{-1}$
Thermo-Physical	Specific heat capacity (J/kg oC) Cure Shrinkage CTE of Composite x 10-6 / oC Resin Degradation Temperature	$Cp = 0.6589 + 0.0025*T$ $csc22 = csc33 = -18.0635\ \alpha^5 - 40.6545\ \alpha^4 + 25.026\ \alpha^3 - 2.50645\ \alpha^2 - 2.6813\alpha$ $CTE = 1.677E-8*T^2 - 1.314E-6*T + 9.439E-5$ 250 oC

Rheological	Viscosity (From Equation 5.6)	A_μ = 6.5452E-12 Pas E_μ = 90477.8 J K/mol α_{Gel} = 0.533 A = 3.185 B = 2.359
	Gel Point	α = 0.35

Table 6.1: Summary Table of Material Properties for Hexcel F155 with Toho T300 Fibers

6.2 Cure Cycle Optimization

In general, the optimization runs have proved that the optimizer is capable of optimizing cure cycles for a composite part, which is representative of a geometry found in composite parts produced in industry. The results of all the optimization runs were presented and discussed in Chapter 5 and summarized in Table B.2 and B.3.

6.2.1 Recommended Configuration for the Optimizer

From the summary in Section 5.2.2.6, the following is the recommend settings for the optimizer:

In terms of general configuration, the optimizer should be run

- Using Micro-GA mode
- Using both the manufacturer and high ramp rate cycles
- Using cure fitness function version 2
- Using time fitness function version 3

In terms of inputs that are specific to GAINPT.dat, the following options should be set to the suggested values:

- Maximum heating rate - 18°F/min
- Uses a high value for PMUT – 0.9
- Number of micro-cycles (NMGA) - 40 while NPOP and NGEN are each set at 10 to give 400 populations

Table 6.2 shows these suggested settings in GAINPT.dat.

```
'Number of members in a population NPOP ..... ', 10
'Number of generation NGEN ..... ', 10
'Number of micro-GA cycles NMGA ..... ', 40
'Apply fitness normalization? (y/n) ..... ', 'n'
'Probability of cross-over PCROSS ..... ', 0.8
'Probability of mutation PMUT ..... ', 0.9
'Allowable X-variations (minutes) XLIM ..... ', 15.
'Allowable Y-variations (deg F) YLIM ..... ', 15.
'Random number seed ..... ', 3
'Number of segments in a curing process NSEG ..... ', 7
'Maximum time for curing (minutes) TIMEMAX ..... ', 310.
'Min allow temp. where process terminates TMIN (deg F) ..... ', 80.
'Max autoclave air temperature TMAX (deg F) ..... ', 430.
'Initial temperature T0 (deg F) ..... ', 80.
'Maximum heating rate (positive, deg F per minute) ..... ', 18
'Maximum cooling rate (negative, deg F per minute) ..... ', -18.
'Location of the fitness output file from Fgen C++ routine .. ',
"C:\APCS\OUTPUTS\fittotal.out"
'Location of the fgen module..... ',
"C:\APCS\fgen\wholeproject.exe"
```

Table 6.2: Suggested Configuration for GAINPT.dat

6.3 Recommendations

The following is a list of suggestions for possible improvements for the optimizer:

- Complete material characterization of Hexcel F155 with Toho T300 fibers for use with the process model when a suitable warpage model is developed.
- Optimize cycles for thicker parts so that the cure and temperature gradient components of the fitness equations can be used and better understood. The composite part used in this thesis was too thin to cause significant gradients.
- Re-work the fitness function code in order to make it run faster. The source code was written by inexperienced programmers and this is reflected in the run time. Currently, the fitness function generator requires more time to run than the process model if only the thermochemical module is turned on.
- Include in COMPRO, or other process models, the ability to output into a file the maximum cure and temperature gradients so that the fitness function does not have to do long searches through all the output data. This will significantly speed up optimization.
- Develop models for residual stress and warpage. The optimizer will be more useful if these modules are included.
- Reprogram the GA so that 2 cool down segments can be used – this should result in faster cool down rates and therefore shorten cure cycle times.
- Remove the seed value used by the random number generator and use the system clock to get a seed number as required. This will ensure true randomness from run to run.

The fitness function, as recommend in Section 6.2.1, is an excellent start towards on-line optimization. Fitness of optimal cycles does improve over the manufacturer's recommended cycle and all cycles achieve minimum requirements in a shorter period of time.

It is expected that the optimizer and material data provided in this thesis, along with SIMCLAVE developed by Michael Hudek (9), will be an excellent starting point for the APCS. It is hoped that one day the entire APCS will become a reality and that this worked helped to lay a useful foundation for the project.

REFERENCES

1. J. M. Kenny, "Integration of Process Models with Control and Optimization of Polymer Composites Fabrication", *Proceedings of the Third Conference on Computer Aided Design in Composite Materials Technology* (1992), pages 530-544.

2. V. Pillai, A.N. Beris and P. Dhurjati, "Intelligent Curing of Thick Section Composites Using a Knowledge-Based System", *Journal of Composite Materials* 31(1) (1997), pages 22-51.

3. P. Ciriscioli, G. Springer, and W. Lee, "An Expert System for Autoclave Curing of Composites", *Journal of Composite Materials* 25 (1991), pages1542-1587.

4. J. Choi and D. Lee, "Expert Cure System for the Carbon Fiber Epoxy Composite Materials", *Journal of Composite Materials* 29(9) (1995), pages1181-1200.

5. N. Rai and R. Pitchumani, "Optimal Cure Cycles for the Fabrication of Thermosetting-Matrix Composites", *Polymer Composites* 18(4) (1997), pages 566-581.

6. S. R. LeClair and F. Abrams, "Qualitative Process Automation", *Proceedings of the IEEE Conference on Decision and Control Including the Symposium on Adaptive Processes* 1(3) (1988), pages 558-563.

7. A. Johnston, "An Integrated Model of the Development of Process-Induced Deformation in Autoclave Processing of Composite Structures", Ph.D., University of British Columbia (1997).

8. M. Hudek, Matt Shewfelt, R. Mark Shead, Andrew Johnston, Nezih Mrad, Rick Cole, J. Raghavan and L. Hendrickson, "Examination of Heat Transfer in Autoclaves", *Proceedings from SAMPE Conference* (2001).

9. M. Hudek, "Examination of Heat Transfer During Autoclave Processing of Polymer Composites", MSc., University of Manitoba, 2001.

10. Hexcel F155 Resin System Information from Hexcel Corportation Website: http://www.hexcelcomposites.com/downloads/techdata/PRE_F155.PDF.

11. C. William Lee and Frances L. Abrams, "Knowledge Base for Expert System Process Control/Optimization." *Achievement in Composites in Japan and the United States. Proceedings Japan-U.S. CCM-V, Tokyo* (1990).

12. A. Trivisiano, J. M. Kenny and L. Nicolais, "Control and Optimization of Autoclave processing of High Performance Composites." *Proceedings of the 37th SAMPE Symposium* (1992), pages 1104-1116.

13. L. T. Biegler and J. E. Cuthrell, "Improved Infeasible Path Optimization for Sequential Modular Simulators II: Optimization Algorithm", *Computer and Chemical Engineering* (1985).

14. S. Vasantharajan and L. T. Biegler, "Large Scale Decomposition for Successive Quadratic Programming", *Computer and Chemical Engineering* (1988).

15. N. Rai and R. Pitchumani, "Optimal Cure Cycle for the Fabrication of Thermosetting-Matrix Composites", *Polymer Composites*, 18(4) (1997), pages 566-581.

16. P. Ciriscioli, Q. Wang and G. Springer, "Autoclave Curing – Comparison of Model and Test Results", *Journal of Composite Materials* 26(1) (1992), page 90.

17. D. Albin, and J. Coulter, "NEROCLAVE: Neural Network Based Intelligent Control of the Autoclave Process", LeHigh University Website: http://www.lehigh.edu/~inmem/thermo/research/374dca.html.

18. M. K. Telkicherla, M. C. Altan and F. C. Lai, "Autoclave Curing of Thermosetting Composites: Process Modeling for the Cure Assembley", *International Communications in Heat and Mass Transfer* 21(6) (1994), pages 785-797.

19. V. Pillai, A. N. Beris, and P. Dhurjati, "Heuristics Guided Optimization of a Batch Autoclave Curing Process", *Computers in Chemical Engineering* 20(3) (1996), pages 275-294.

20. D. Tse and L. Chan, "Application of Micro Genetic Algorithms and Neural Networks for Airfoil Design Optimization", *Nato RTO/AVT Symposium on Aerodynamic Design and Optimization of Flight Vehicles in a Concurrent Multi-Disciplinary Environment* (1999), pages 23-1 – 23-11.

21. W. Chunfeng, Q. Hongyin and X. Xien, "Optimal Design of Multiproduct Batch Chemical Process Using Genetic Algorithms", *Industrial and Engineering Chemistry Research* 35(10) (1996), pages 3560-3566.

22. R. Mathur, B. Fink and S. Adavani, "Use of Genetic Algorithms to Optimize Gate and Vent Locations for the Resin Transfer Molding Process", *Polymer Composites*, 20(2) (1999), pages 167-178.

23. A.C. Loos and G.S. Springer, "Curing of Epoxy Matrix Composites", *Journal of Composite Materials* 17(2) (1983), pages 135-169.

24. S.R. White and H.T. Hahn, "Mechanical Property and Residual Stress Development During Cure of a Graphite/BMI Composite", *Polymer Engineering and Science* 30(22) (1990).

25. T.A. Bogetti and J.W. Gillespie Jr., "Two-dimensional Cure Simulation of Thick Thermosetting Composites", *Journal of Composite Materials* 25(3), (1991), pages 239-273.

26. T.A. Bogetti and J.W. Gillespie Jr., "Process-Induced Stress and Deformation in Thick-Section Thermoset Composite Laminates", *Journal of Composite Materials* 26(5) (1992), pages 626-660.

27. S.R. White and H.T. Hahn, "Process Modeling of Composite Materials: Residual Stress Development During Cure Period Part II. Experimental Validation", *Journal of Composite Materials* 26(16) (1992), pages 2423-2453.

28. S.R. White and H.T. Hahn, "Cure Cycle Optimization for the Reduction of Processing-Induced Stresses in Composite Materials", *Journal of Composite Materials* 27(14) (1992), pages 1352-1378.

29. M. Telikicherla, X. Li, M. C. Altan and F. C. Lai, "Numerical Study of Conjugate Heat Transfer in Autoclave Curing of Thermosetting Composites", *HTD-Vol. 289, ASME* (1994), pages 213-221.

30. N. Ghariban, D. Lou, A. Haji-Sheikh, "The Effect of Honeycomb Flow Straighteners on Turbulence and Heat Transfer in an Autoclave Model", *HTD-Vol.* 233, *ASME* (1992), pages 45-52.

31. A. Johnston, P. Hubert, Reza Vaziri and Anoush Poursartip, "An Investigation of Autoclave Convective Heat Transfer", *Proceedings of the Second Joint Canada-Japan Workshop on Composites* (1998).

32. Koteshwara, M. "Parametric Study of Process-induced Warpage in Composite Laminates." MSc., University of Manitoba, (2001).

APPENDIX A

Material Characterization Data

A.0 Introduction

This appendix contains various plots from the material characterization experiments.

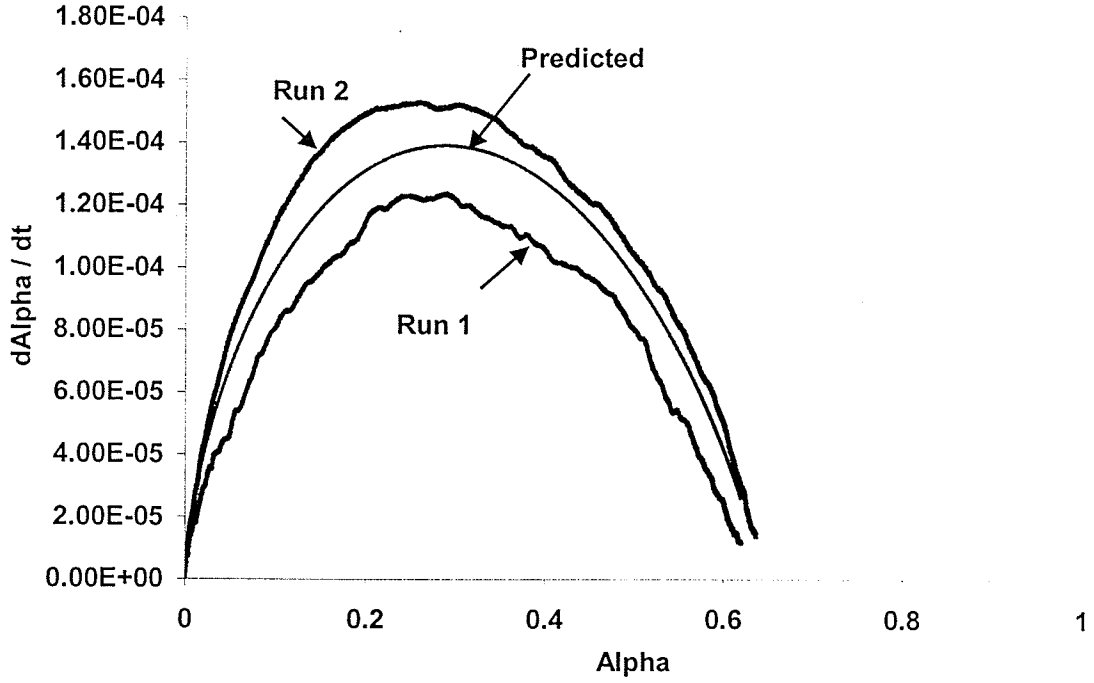

Figure A.1: Plot of α versus $d\alpha/dt$ for Isothermal Run at 90°C

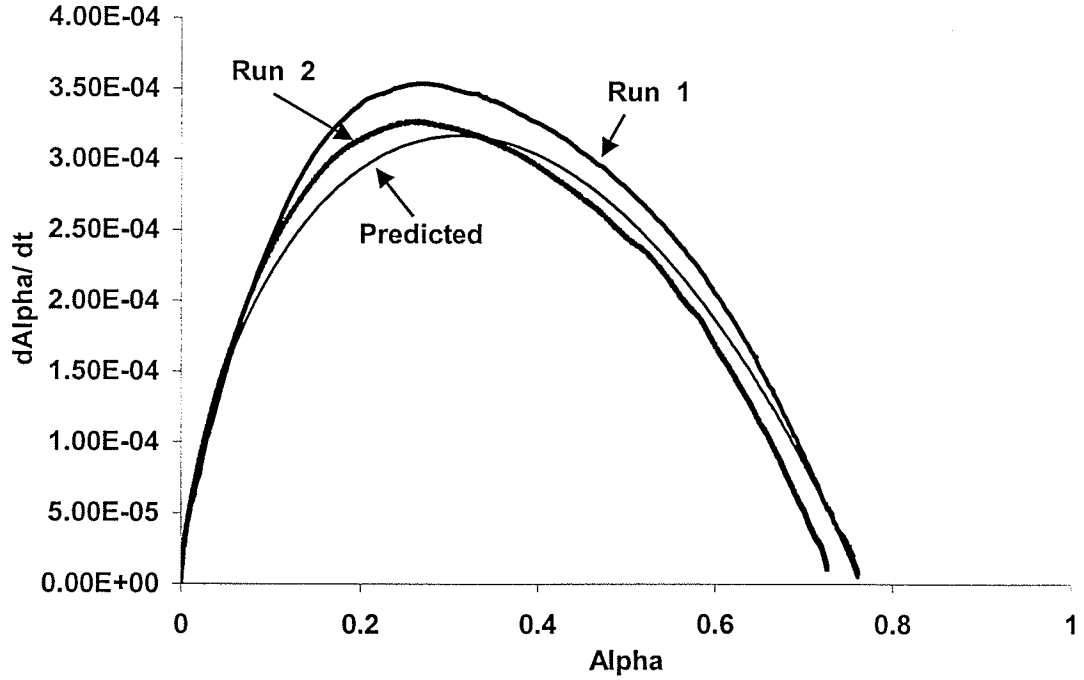

Figure A.2: Plot of α versus $d\alpha/dt$ for Isothermal Run at 100°C

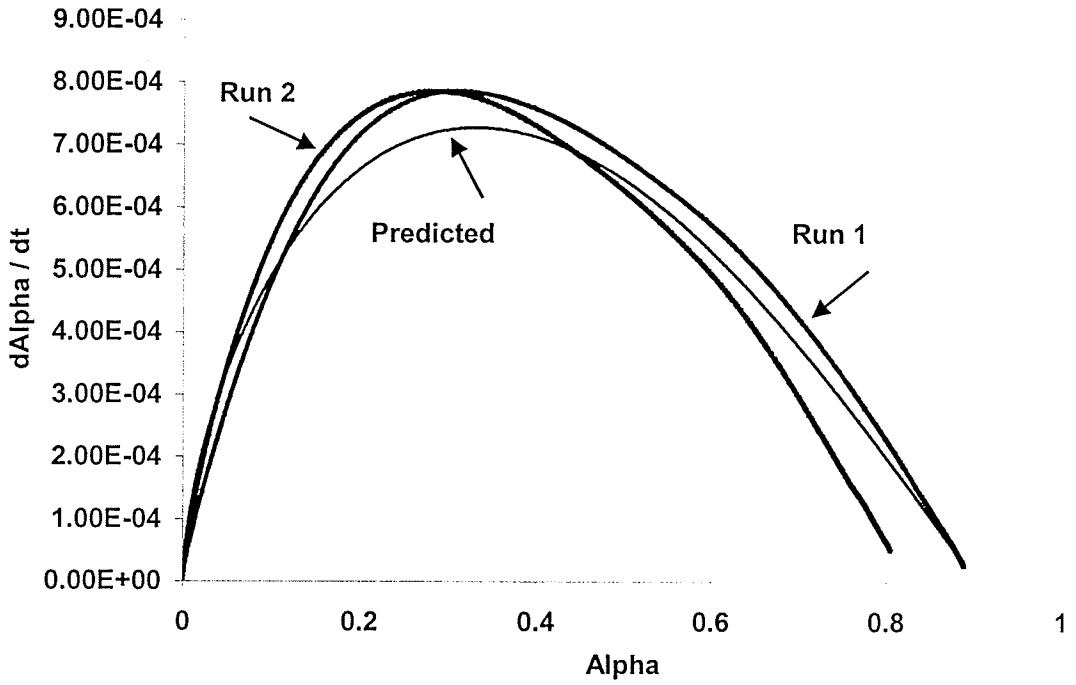

Figure A.3: Plot of α versus $d\alpha/dt$ for Isothermal Run at 110°C

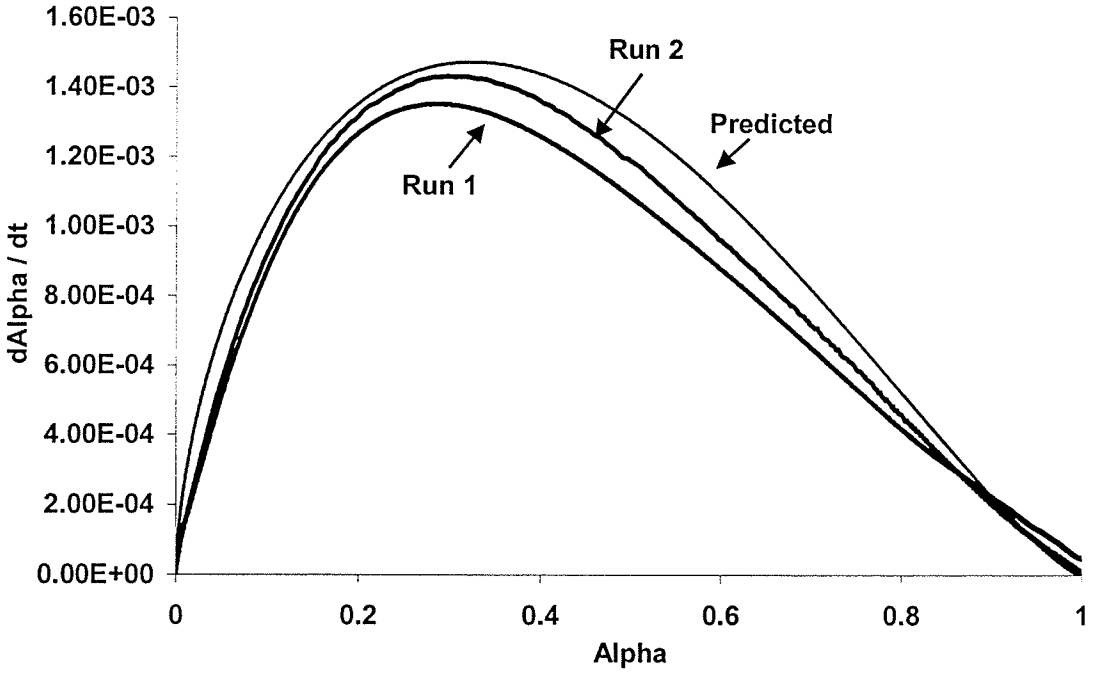

Figure A.4: Plot of α versus $d\alpha/dt$ for Isothermal Run at 120°C

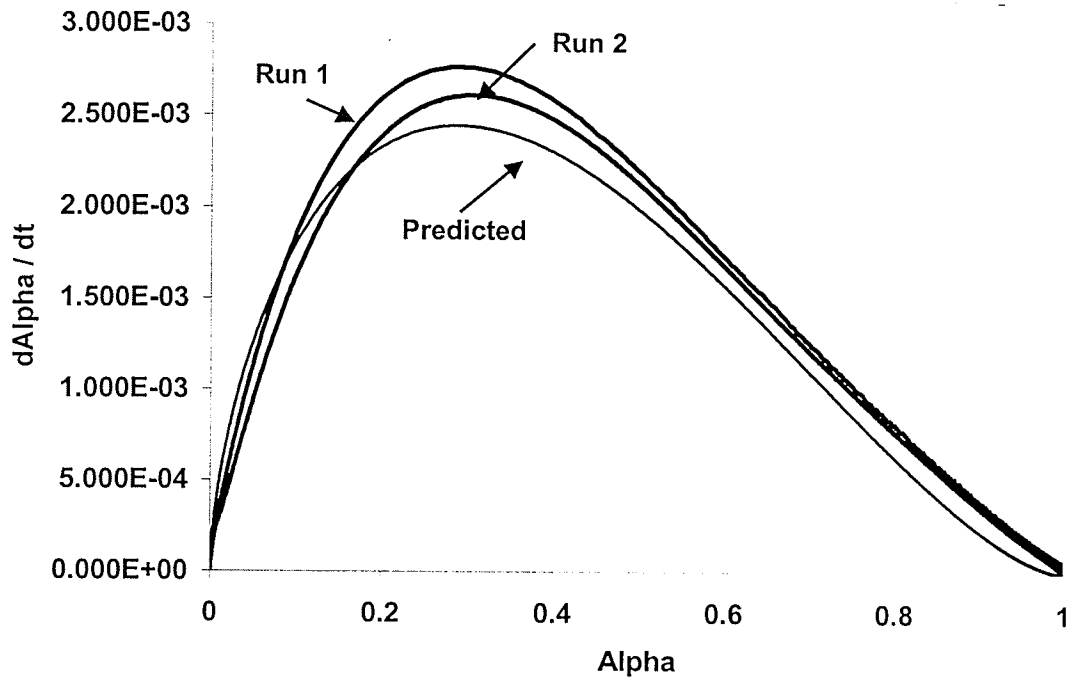

Figure A.5: Plot of α versus $d\alpha/dt$ for Isothermal Run at 130°C

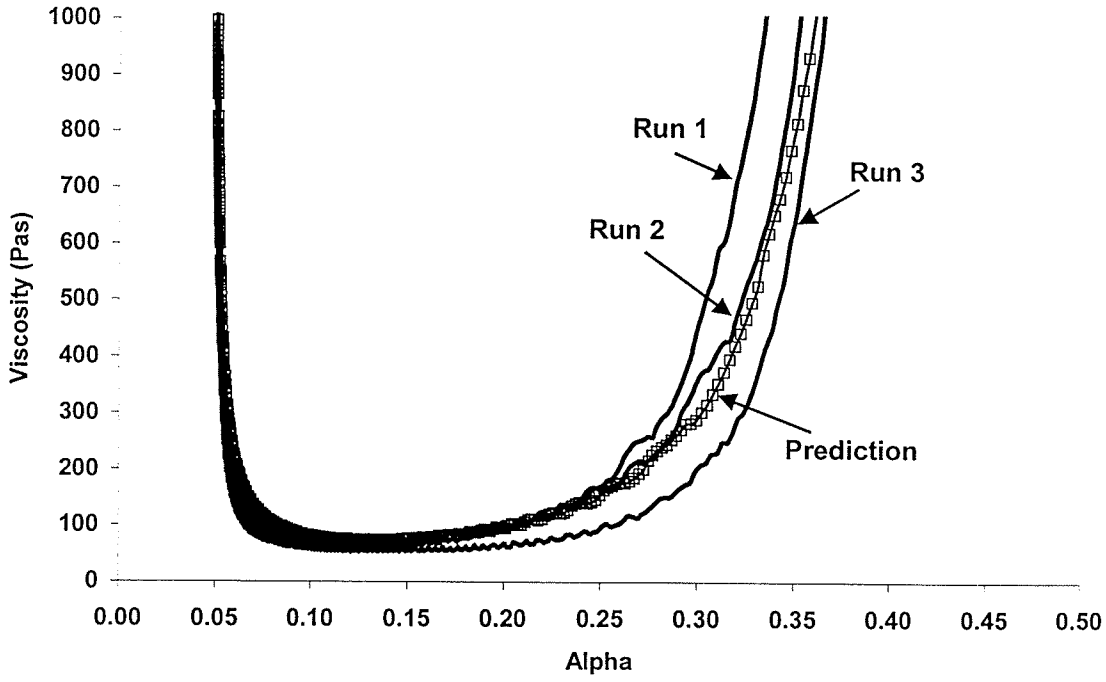

Figure A.6: Plot of Experimental and Predicted Viscosity versus Degree of Cure for a Dynamic Scan of 2°C/min

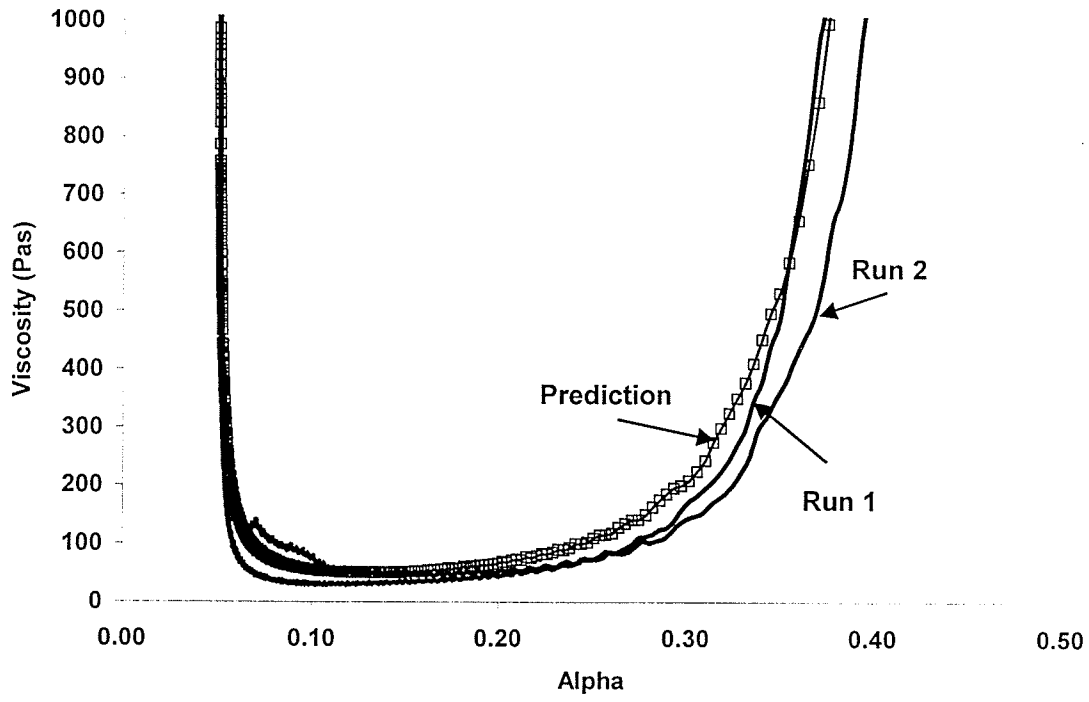

Figure A.7: Plot of Experimental and Predicted Viscosity versus Degree of Cure for a Dynamic Scan of 3°C/min

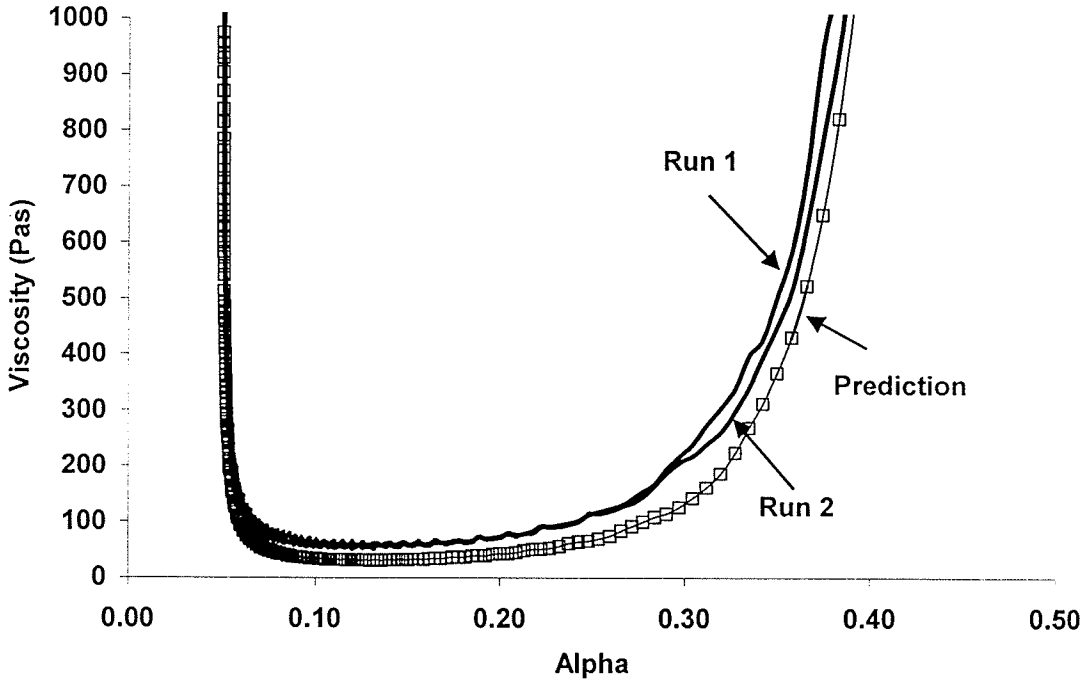

Figure A.8: Plot of Experimental and Predicted Viscosity versus Degree of Cure for a Dynamic Scan of 5°C/min

Appendix B

Optimization Run Results

B.0 Introduction

This appendix contains results from individual optimization runs. The run numbers and various settings are shown in Table B.1. Tables B.2 and B.3 summarize various aspects of the optimization runs. Figures B.1 shows the manufacturer's recommended cycle while Figure B.2 shows the high ramp rate cycle. Figures B.3 and up provide fitness plots and optimal cycles for each of the optimization runs.

The following describes each of the parameters in the Tables B.2 and B.3

Total time –the total cycle time provided in minutes for an optimized cycle

$\mathbf{T_{max}}$ – The maximum temperature at any node in the composite at any time during the cure.

α_{min}– The minumum degree of cure in the composite at the end of the cure cycle

Fitness Temperature - The fitness assigned by the temperature fitness function (a value between 0 and 1)

Fitness alpha – The fitness assigned by the degree of cure fitness function (a value between 0 and 1)

Fitness time – The fitness assigned by the time fitness function (a value between 0 and 1)

FitnessTotal – The total fitness assigned to the cure cycle (summation of fitness alpha, fitness temperature, fitness time as shown above – value between 0 and 3)

Cure Gradient – The maximum degree of cure gradient in the part at any given time during the cure.

Temperature Gradient - The maximum temperature gradient in the part at any given time during the cure.

Run No.	Objective	NPOP	NGEN	NMGA	PMUT	Seed	Max HR (pos)	Input manfac cycle (y/n)	Input High Ramp Cycle	Change Fitness Function
1	Determine number of iterations to convergence	10	2000	1	0.9	3	7.9	n	n	n
2	Determine the effect of random seed	10	400	1	0.9	7	7.9	n	n	n
3	Set baseline of Regular GA	10	400	1	0.9	3	7.9	n	n	n
4	Set baseline of micro GA	10	40	10	0.9	3	7.9	n	n	n
5	Effects of injecting manufacturer cycle (Reg GA)	10	400	1	0.9	3	7.9	y	n	n
6	Effects of injecting manufacturer cycle (Micro GA)	10	40	10	0.9	3	7.9	y	n	n
7	Effects of injecting manufacturer cycle and high ramp rate cycle (Reg GA)	10	400	1	0.9	3	18	y	y	n
8	Effects of injecting manufacturer cycle and high ramp rate cycle (Micro GA)	10	40	10	0.9	3	18	y	y	n
9	Effects of injecting manufacturer cycle with new cure fitness function (version 2) (Reg GA)	10	400	1	0.9	3	7.9	y	n	y
10	Effects of injecting manufacturer cycle with new cure fitness function (version 2) (Micro GA)	10	40	10	0.9	3	7.9	y	n	y
11	Effects of injecting manufacturer cycle and high ramp	10	400	1	0.9	3	18	y	y	y

	rate cycle with new cure fitness function (version 2) (Reg GA)									
12	Effects of injecting manufacturer cycle and high ramp rate cycle with new cure fitness function (version 2) (Micro GA)	10	40	10	0.9	3	18	y	y	y
13	Effects of injecting manufacturer cycle and high ramp rate cycle with new cure fitness function (version 3) (Micro GA)	10	40	10	0.9	3	18	y	y	y
14	Effects of new time function (version 2) and 2 injected cycles with new cure function (version 2) (Micro GA)	10	40	10	0.9	3	18	y	y	y
15	Effects of new time function (version 2) and 2 injected cycles with new cure function (version 3) (Micro GA)	10	40	10	0.9	3	18	y	y	y
16	Effects of PMUT on setup as used in Run 12	10	40	10	0.1	3	18	y	y	y
17	Effect of changing NMGA on setup used in Run12	10	10	40	0.9	3	18	y	y	y
18	Optimal Settings – Cure function (Version 2), Time Function (Version 2), inject 2 cycles, with high NMGA in the Micro-GA mode	10	10	40	0.9	3	18	y	y	y

Table B.1: Summary of Inputs used for Optimization Runs

Run	Total Time (min)	T_{max} (°F)	α_{min}	Fitness Temp	Fitness Alpha	Fitness Time	Total Fitness
1	243	NA	NA	NA	NA	NA	NA
2	242	NA	NA	NA	NA	NA	NA
3	242	NA	NA	NA	NA	NA	NA
4	230	NA	NA	NA	NA	NA	NA
5	279	370	0.958	1	0.870	0.782	2.658
6	277	370	0.960	1	0.876	0.782	2.658
7	284	369.98	0.958	1	0.869	0.766	2.634
8	265	370	0.958	1	0.069	0.782	2.65
9	265	369.96	0.92	1	1	0.822	2.82
10	258	369.98	0.92	1	1	0.830	2.83
11	244	369.99	0.92	1	1	0.814	2.814
12	236	369.97	0.92	1	1	0.822	2.82
13	264	369.64	0.93	0.994	1	0.790	2.78
14	200	369.97	0.92	1	1	0.841	2.84
15	250	369.99	0.935	0.998	1	0.804	2.803
16	267	369.92	0.92	1	1	0.774	2.743

17	203	370	0.92	1	1	0.83	2.830
18	134	370	0.92	1	1	0.83	2.890

Table B.2: Summary of Optimization Results

Run	Cure Gradient ($\Delta\alpha$)	Temp Gradient (ΔT in °F)
1	NA	NA
2	NA	NA
3	NA	NA
4	NA	NA
5	0.015	6.61
6	0.016	6.81
7	0.018	8.62
8	0.018	8.91
9	0.015	7.07
10	0.015	7.59
11	0.020	8.89
12	0.019	8.89
13	0.018	8.89
14	0.017	7.61
15	0.016	7.66
16	0.015	7.84
17	0.018	7.52

Table B.3 – Summary of Results – Temperature and Cure Gradients

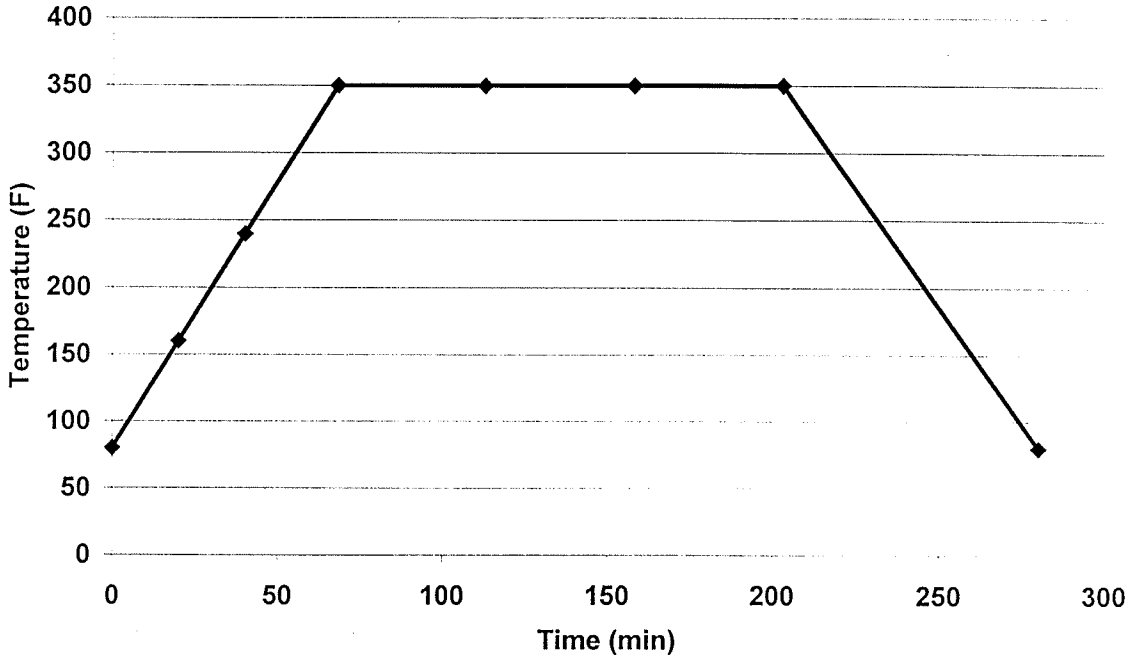

Figure B1: Manufacturer's Recommended Cure Cycle

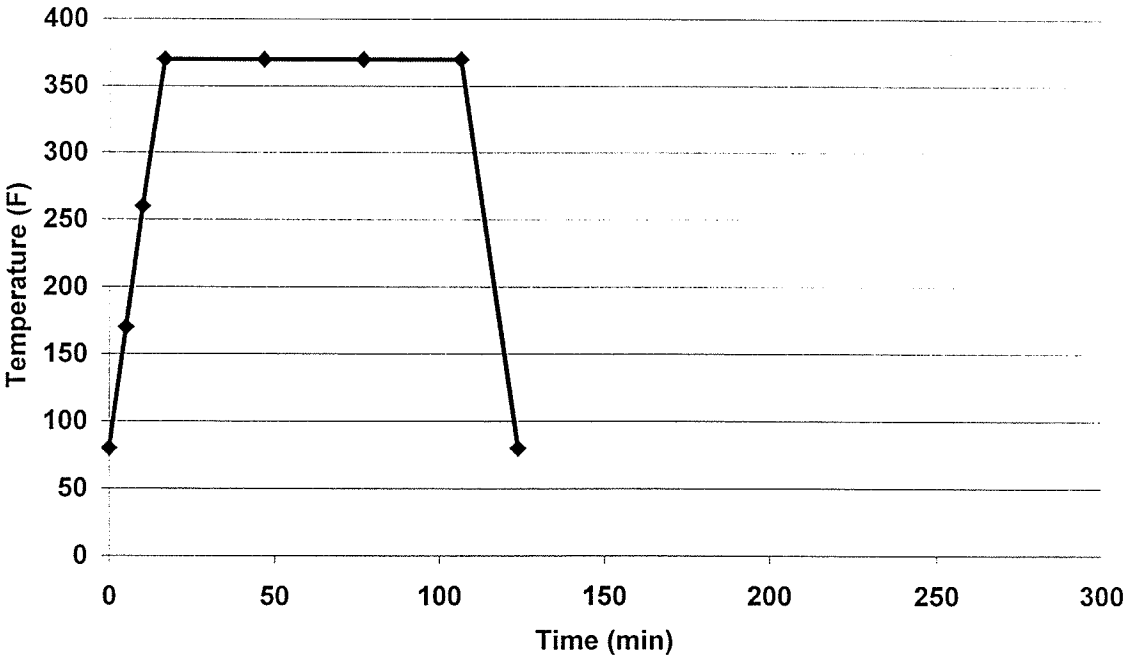

Figure B.2: Plot of High Ramp Rate Cycle

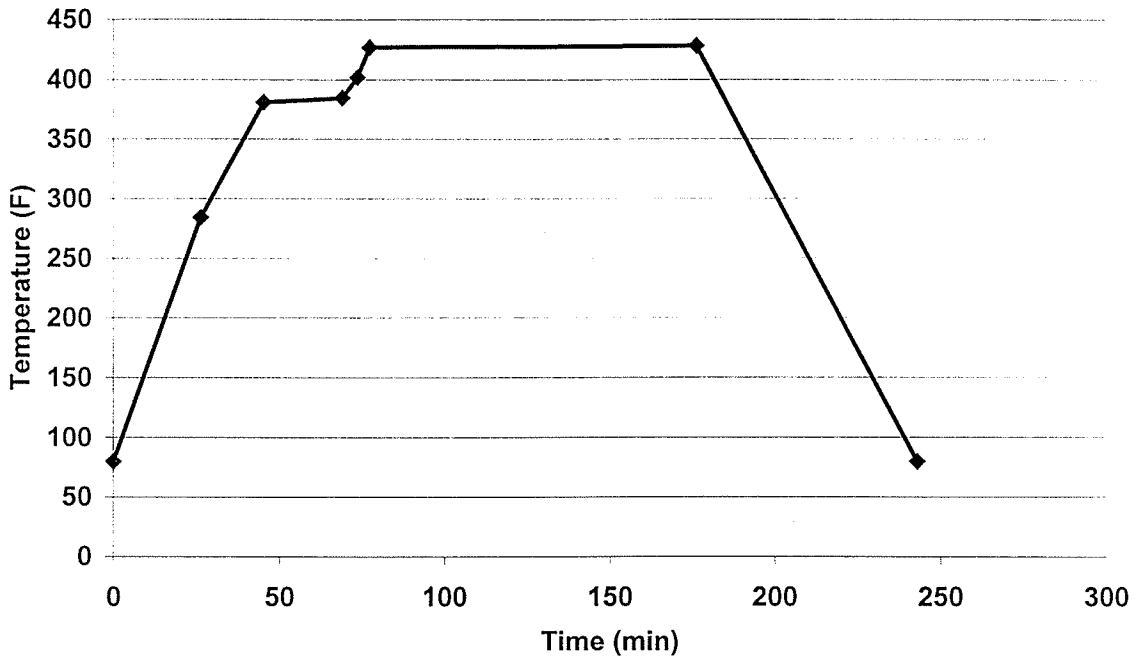

Figure B.3: Plot of Optimal Cycle from Run 1

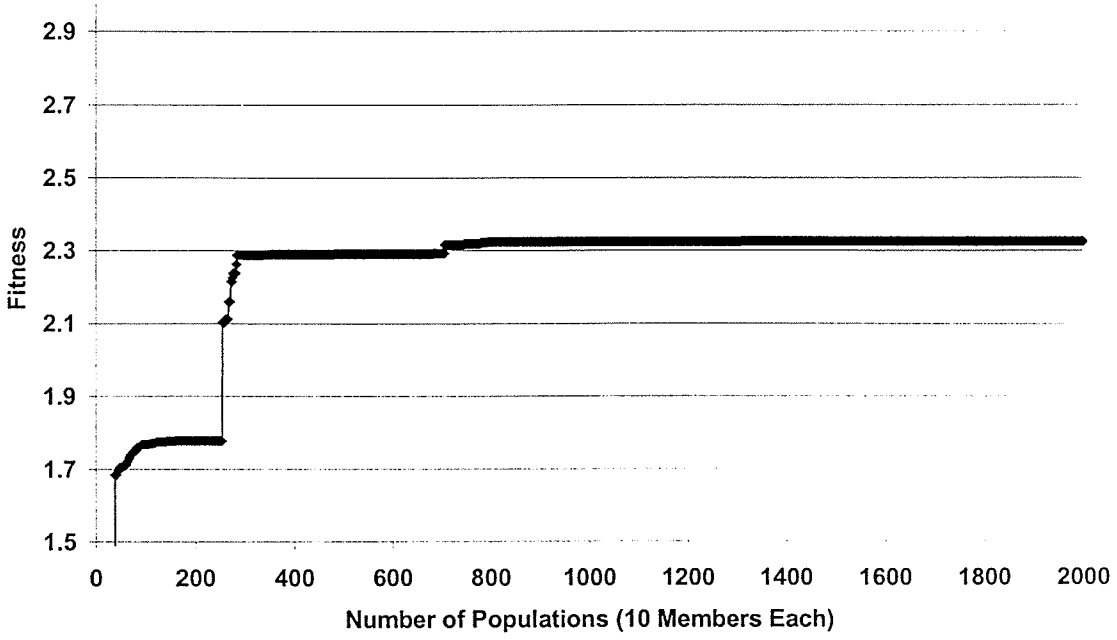

Figure B.4: Plot of Fitness versus Number of Populations from Run 1

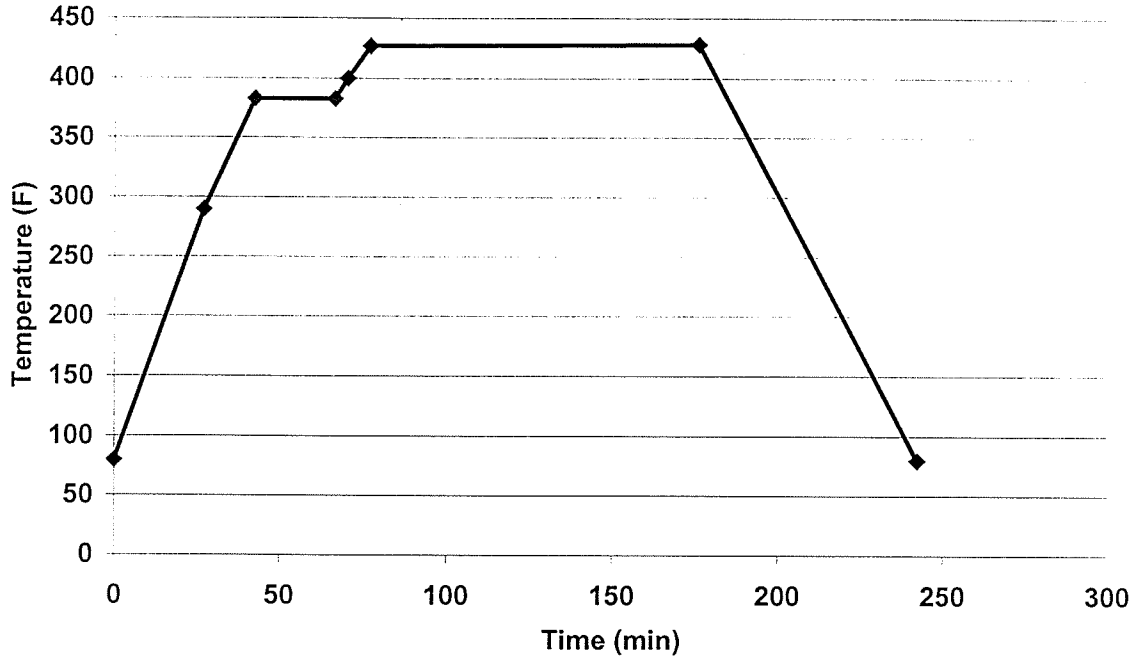

Figure B.5: Plot of Optimal Cycle from Run 2

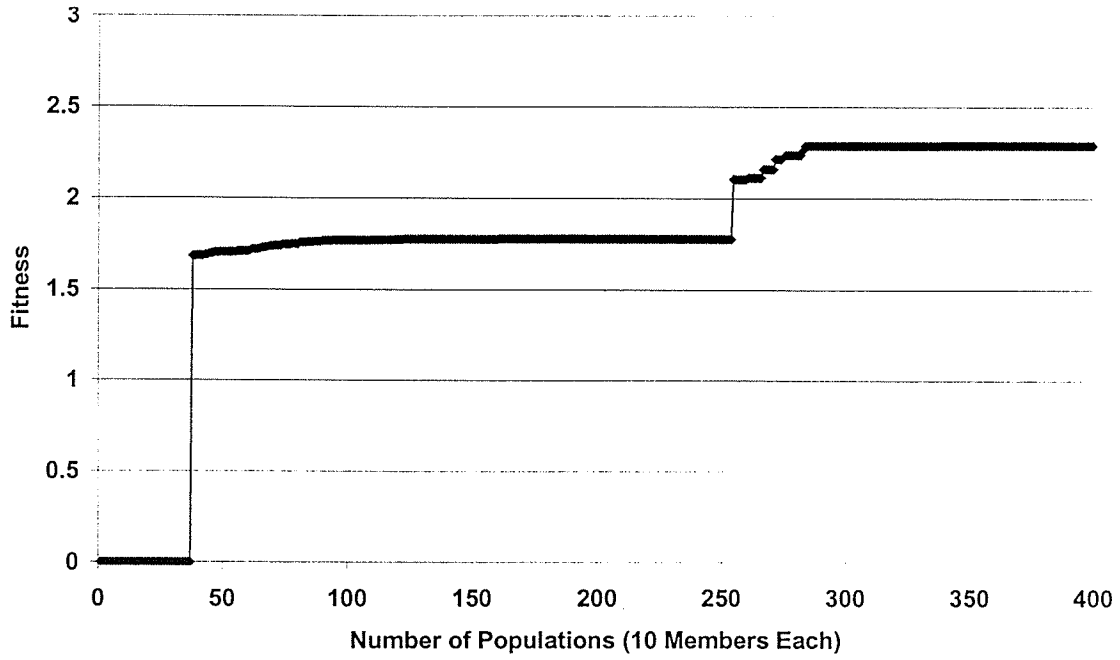

Figure B.6: Plot of Fitness versus Number of Populations for Run 2

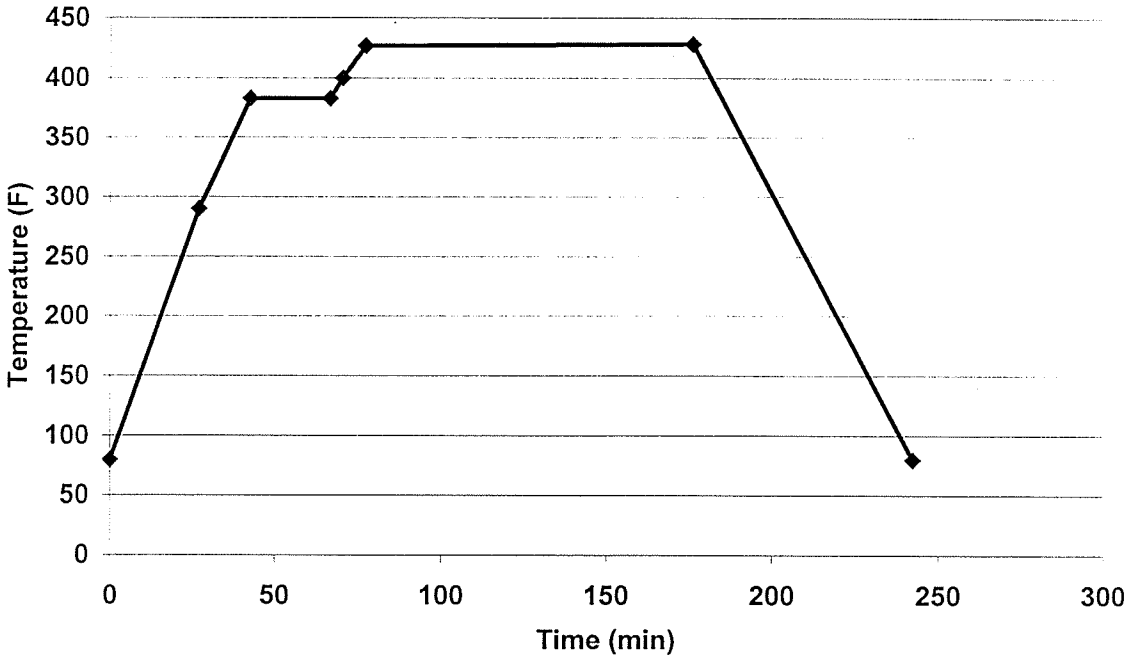

Figure B.7: Plot of Optimal Cycle from Run 3

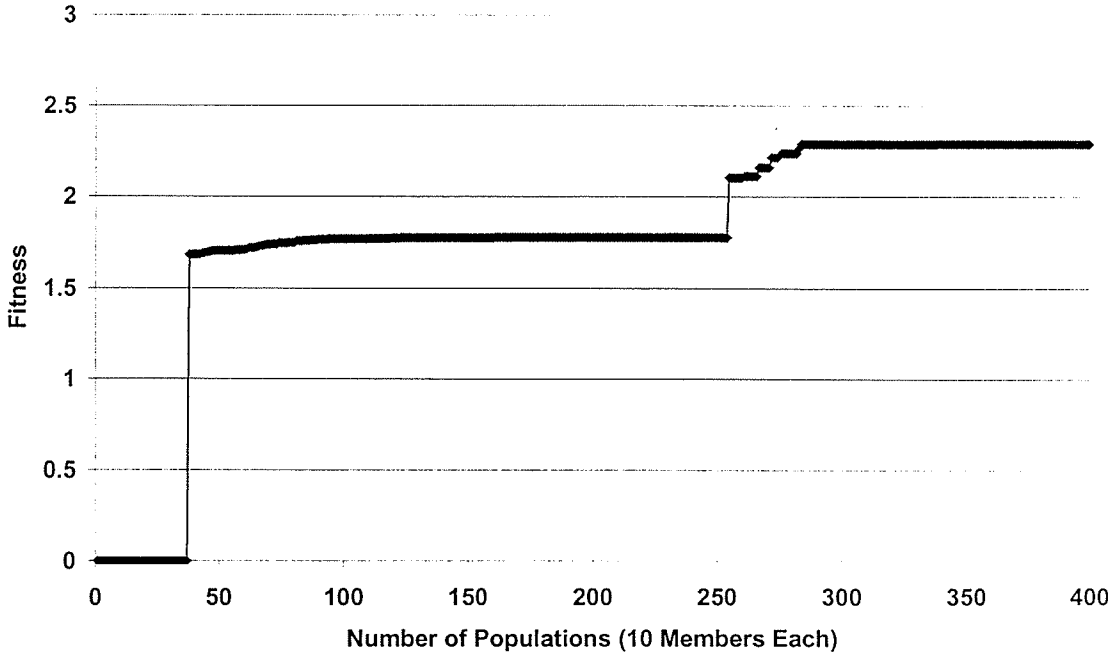

Figure B.8: Plot of Fitness versus Number of Populations for Run 3

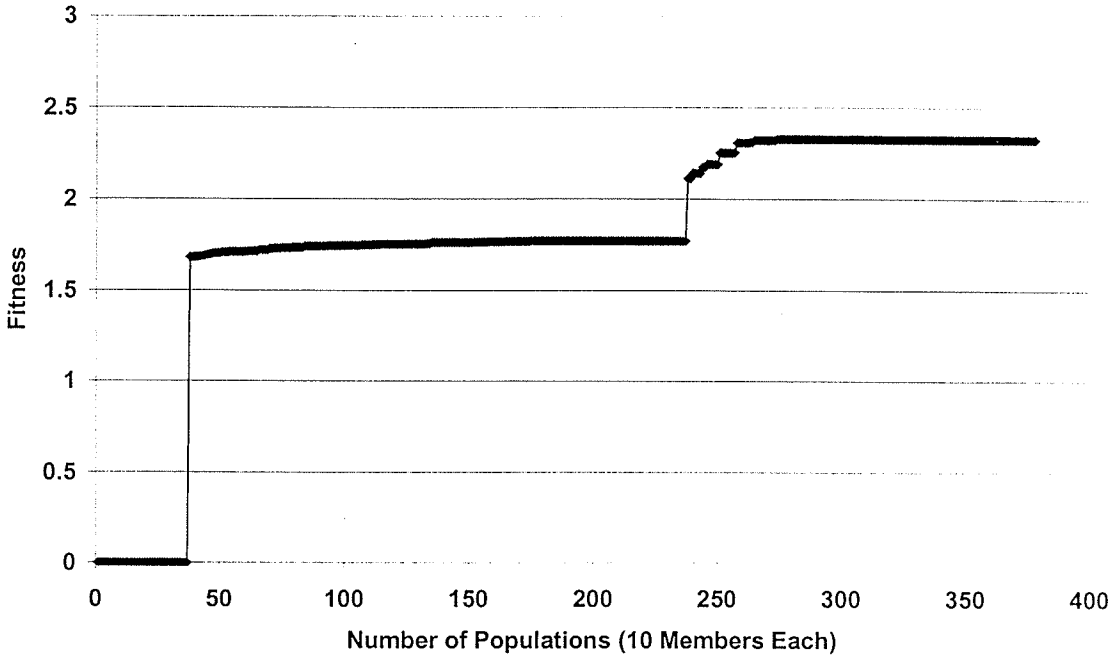

Figure B.9: Plot of Optimal Cycle from Run 4

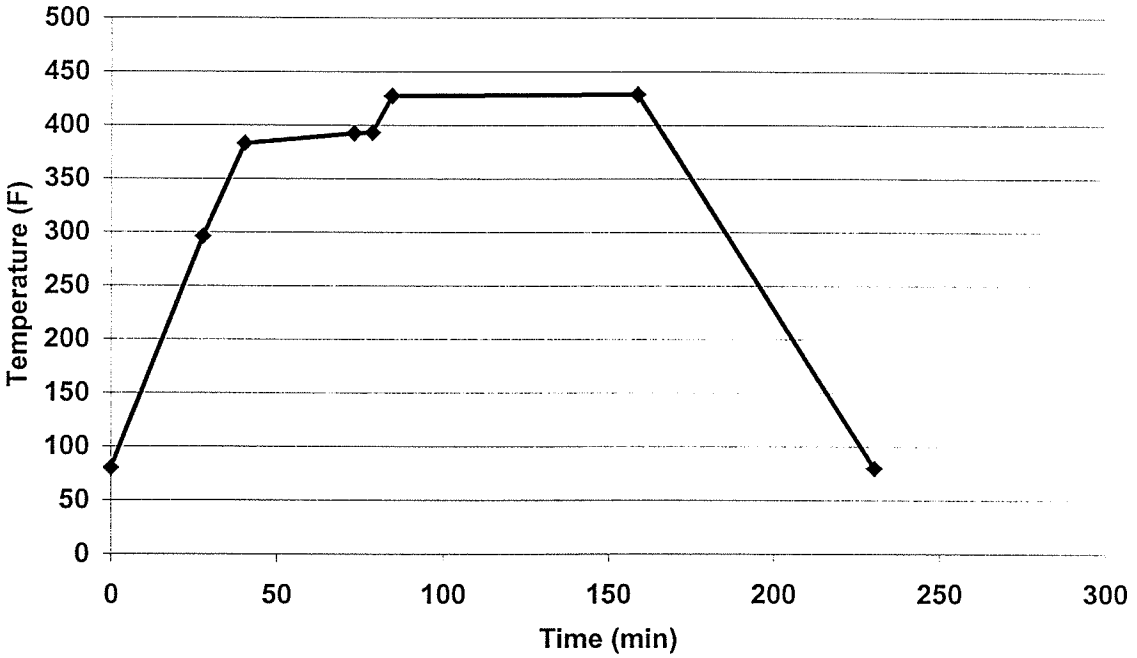

Figure B.10: Plot of Fitness versus Number of Populations for Run 4

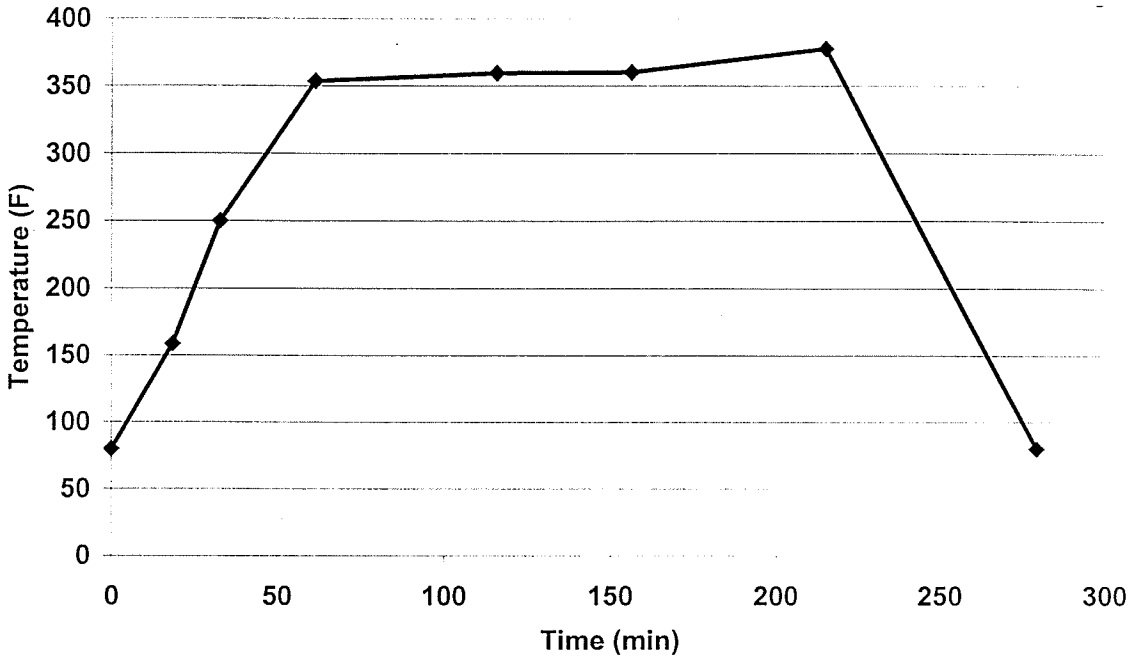

Figure B.11: Plot of Optimal Cycle from Run 5

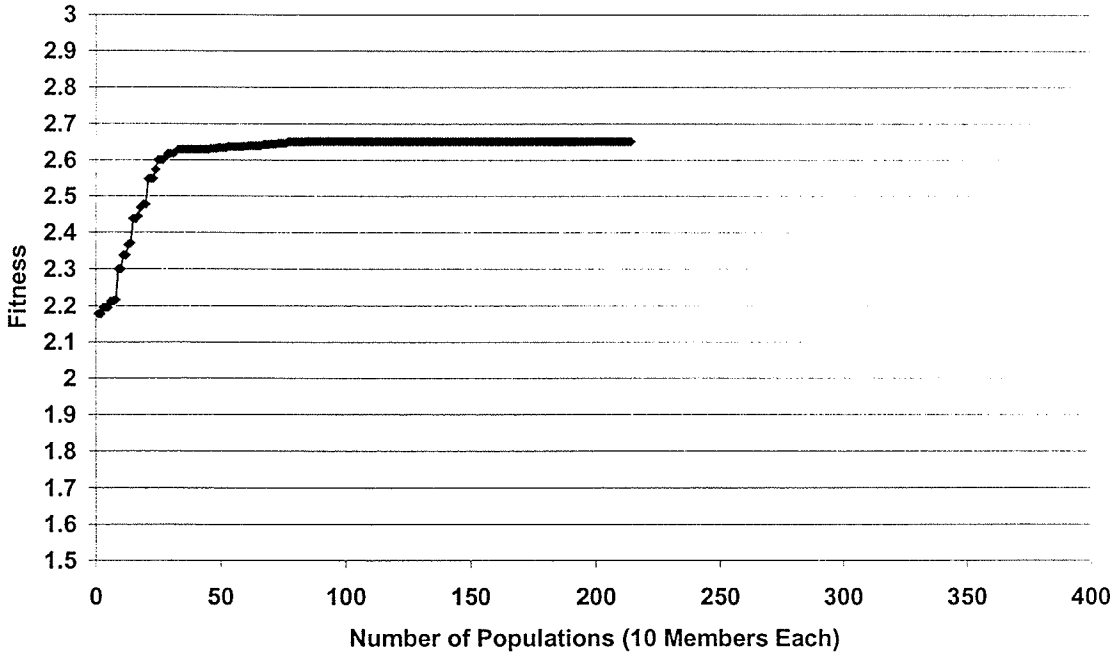

Figure B.12: Plot of Fitness versus Number of Populations for Run 5

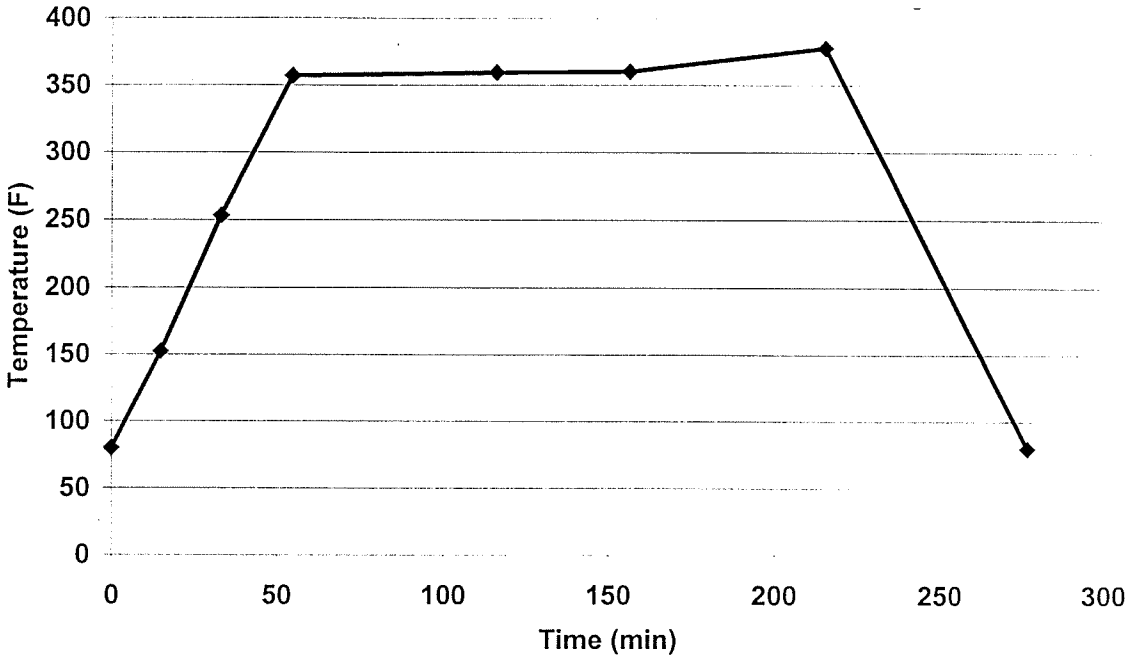

Figure B.13: Plot of Optimal Cycle from Run 6

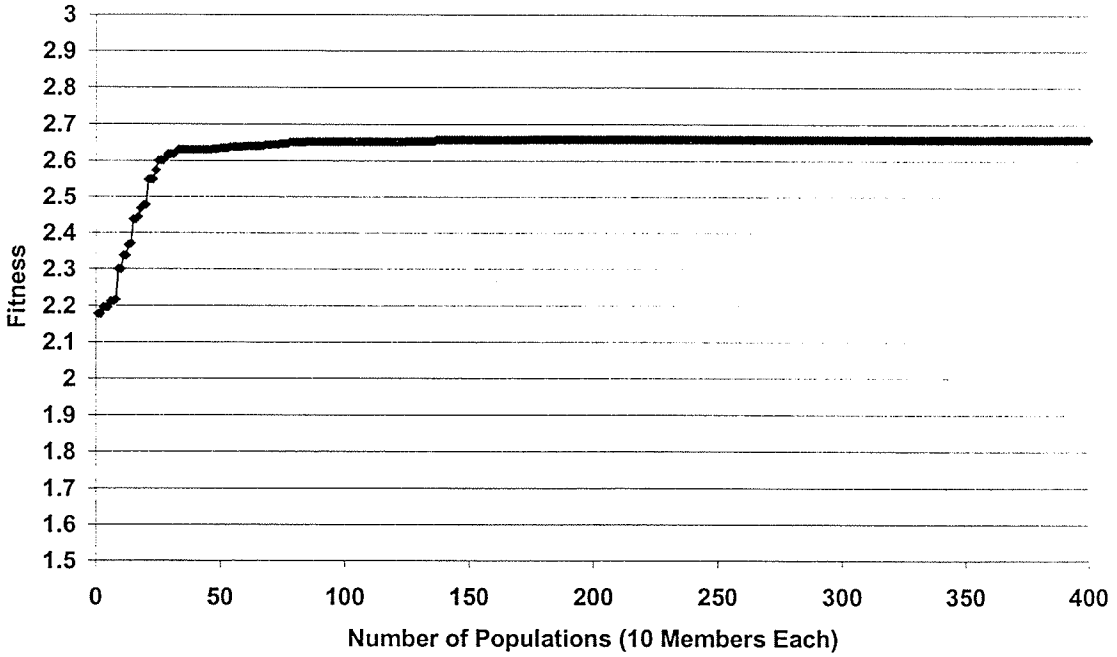

Figure B.14: Plot of Fitness versus Number of Populations for Run 6

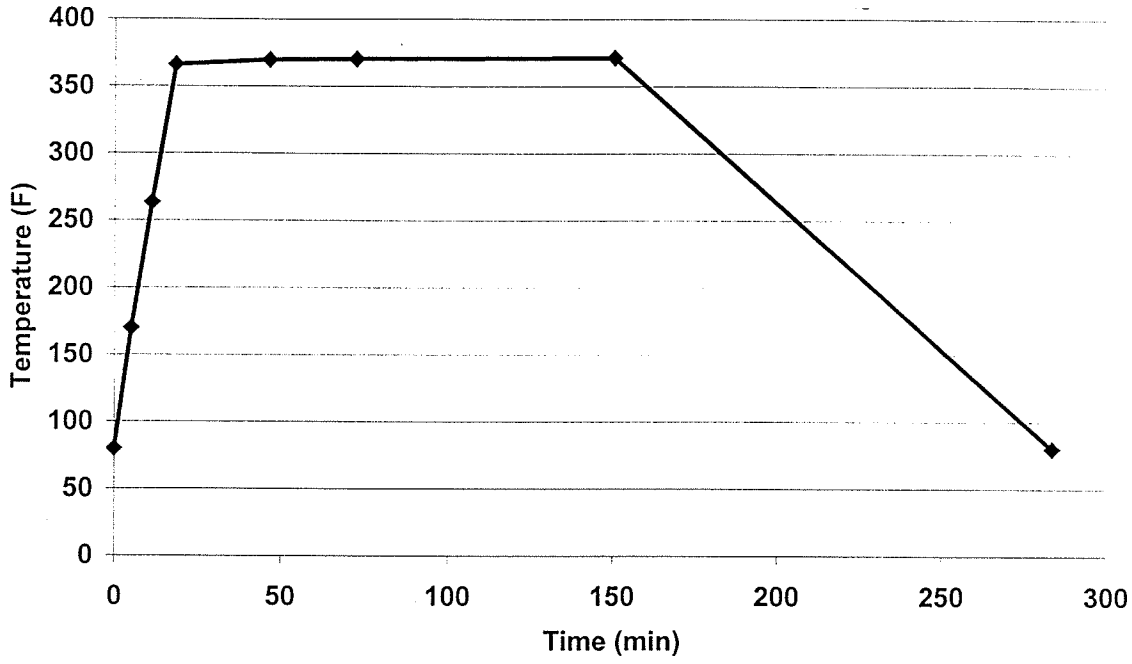

Figure B.15: Plot of Optimal Cycle from Run 7

Figure B.16: Plot of Fitness versus Number of Populations for Run 7

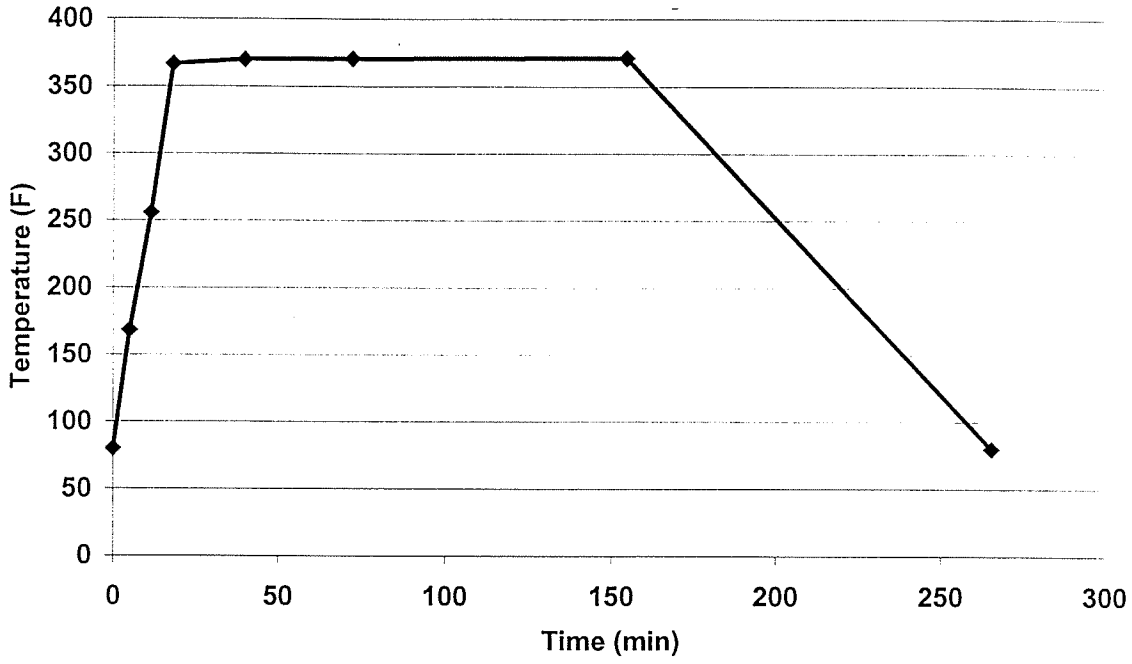

Figure B.17: Plot of Optimal Cycle from Run 8

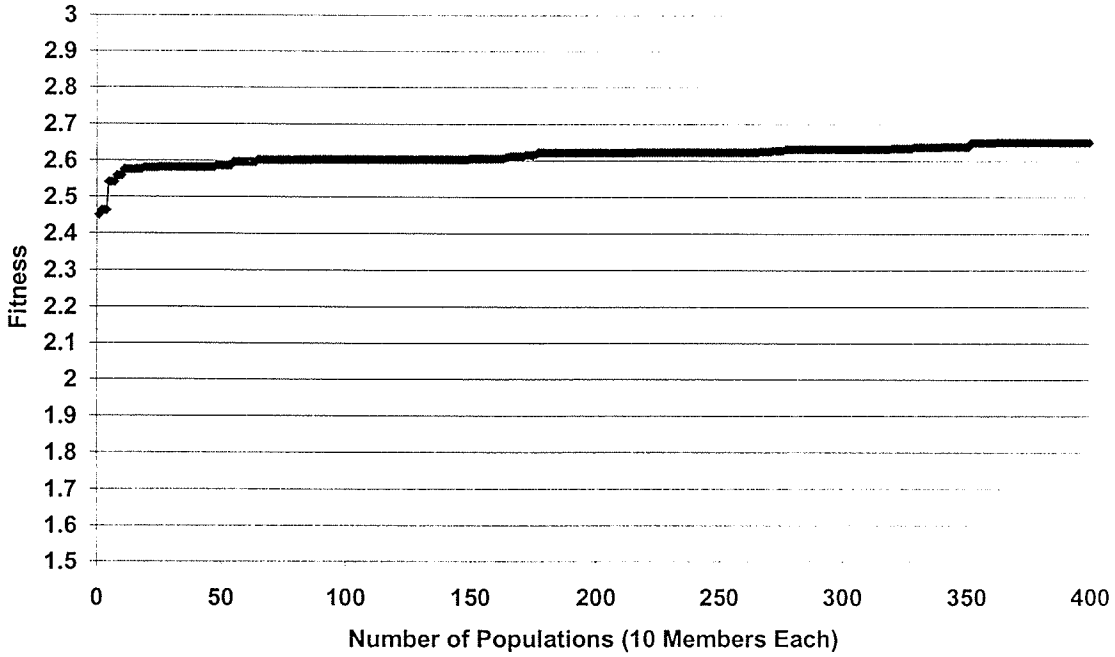

Figure B.18: Plot of Fitness versus Number of Populations for Run 8

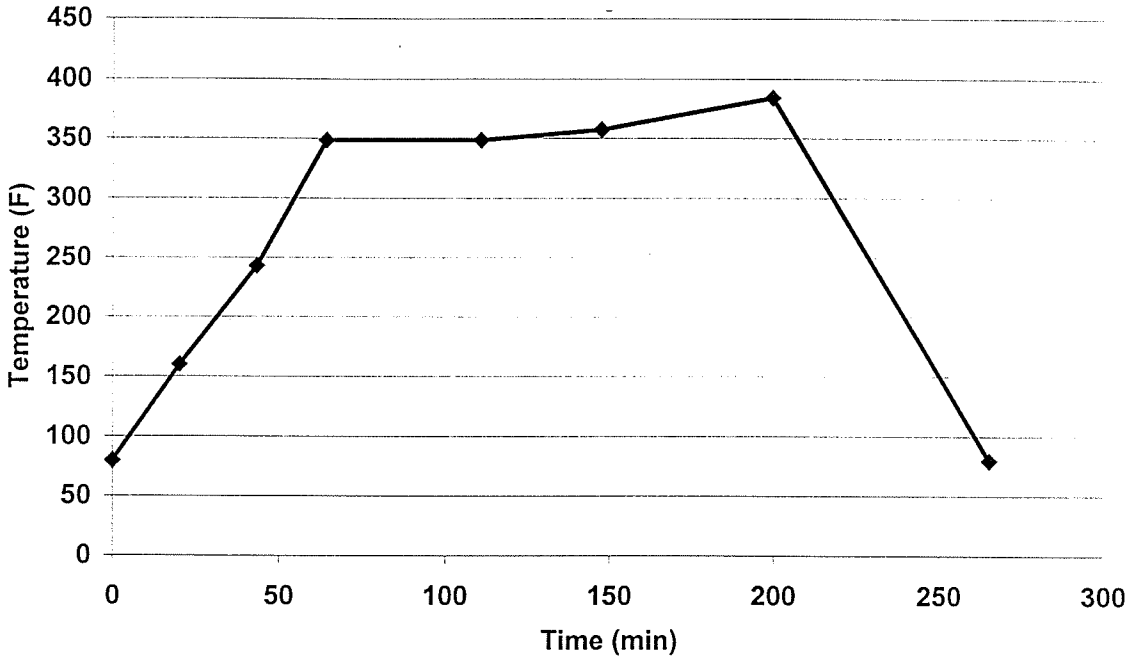

Figure B.19: Plot of Optimal Cycle from Run 9

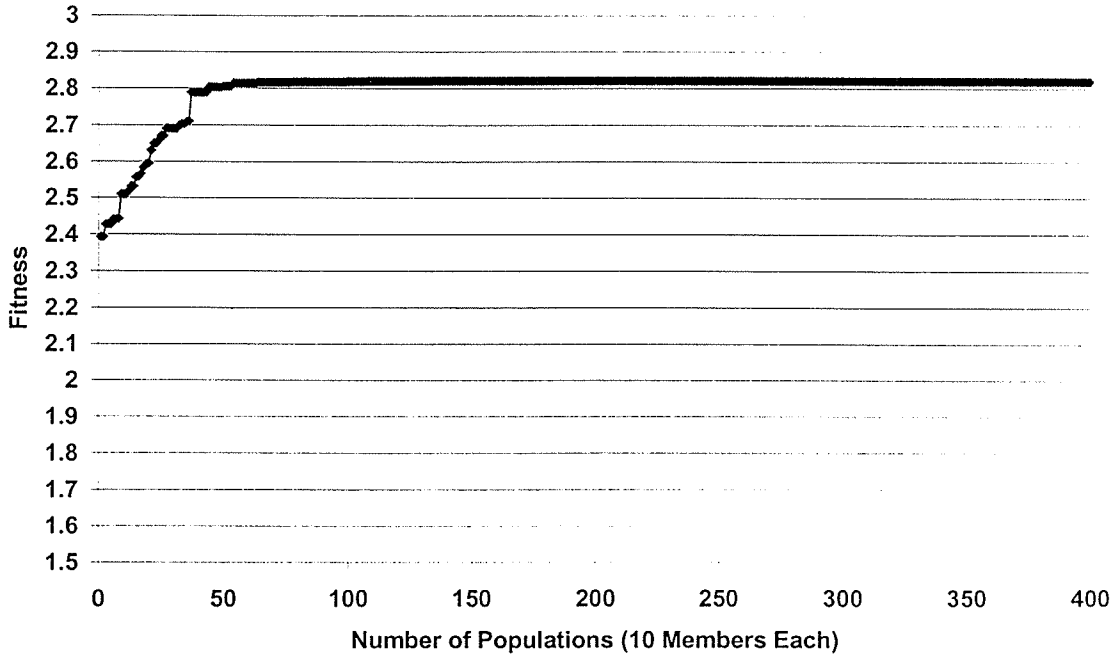

Figure B.20: Plot of Fitness versus Number of Populations for Run 9

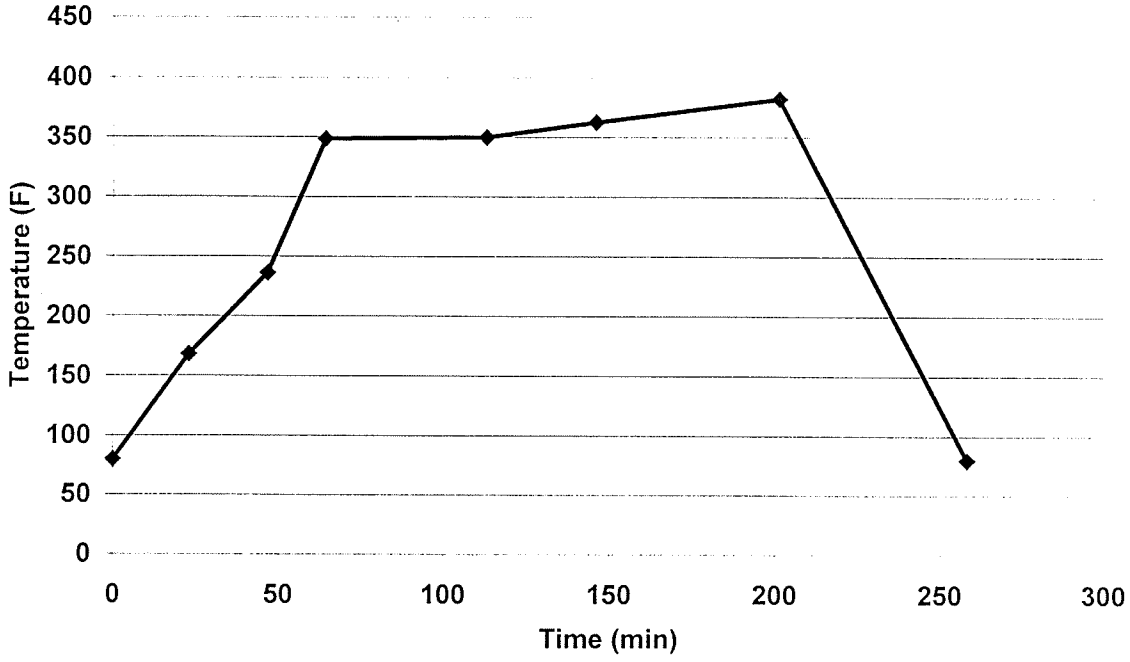

Figure B.21: Plot of Optimal Cycle from Run 10

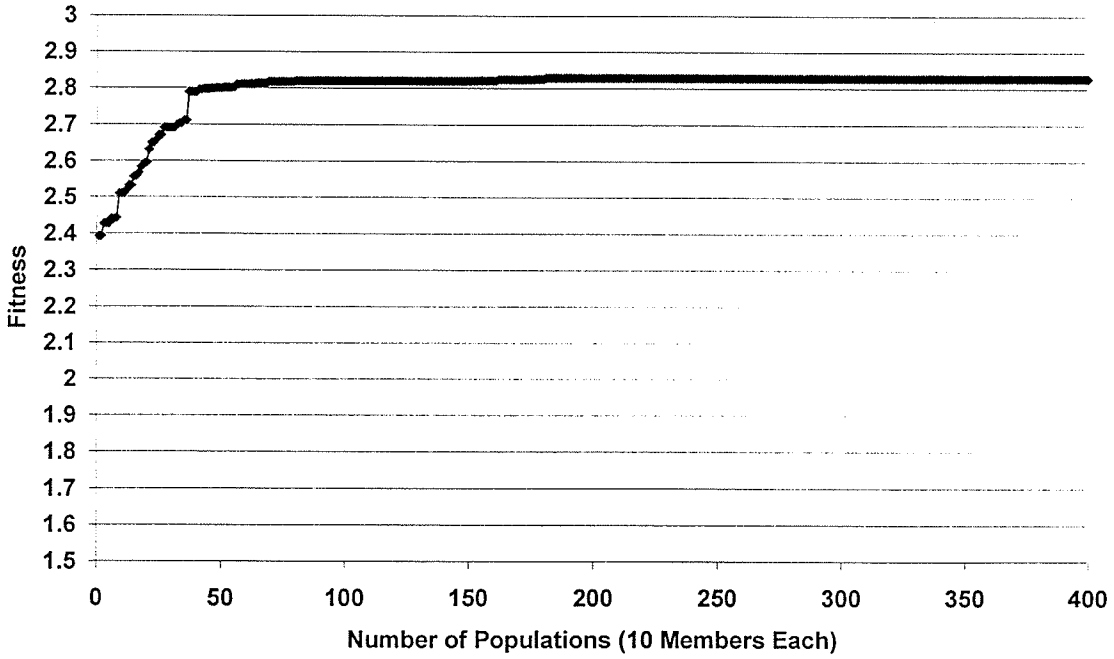

Figure B.22: Plot of Fitness versus Number of Populations for Run 10

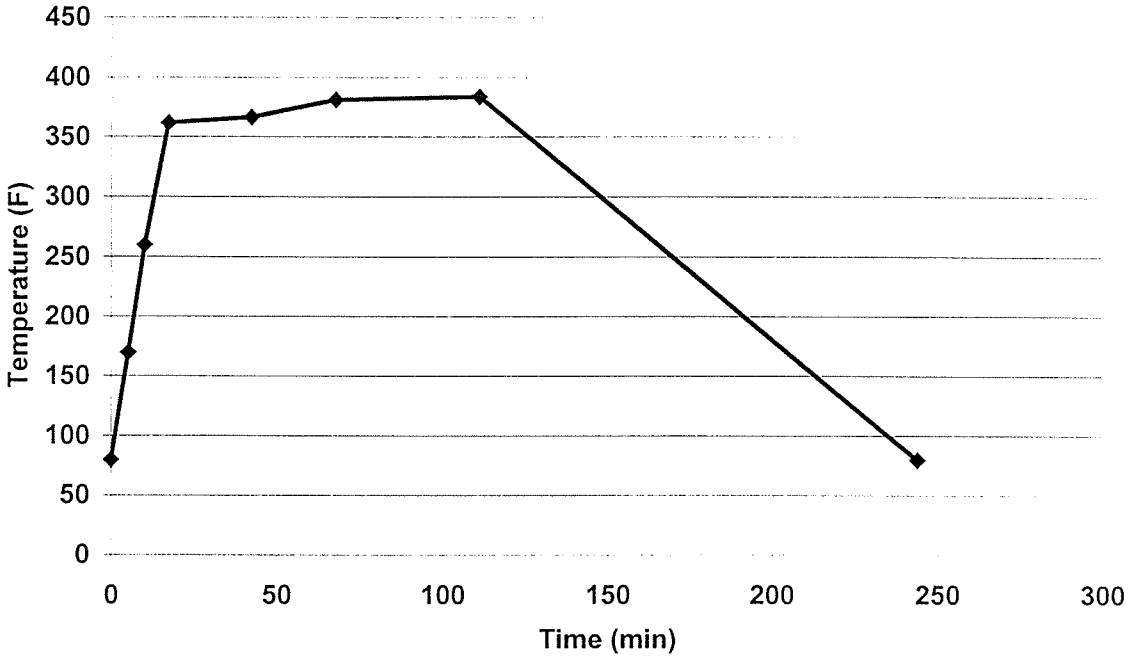

Figure B.23: Plot of Optimal Cycle from Run 11

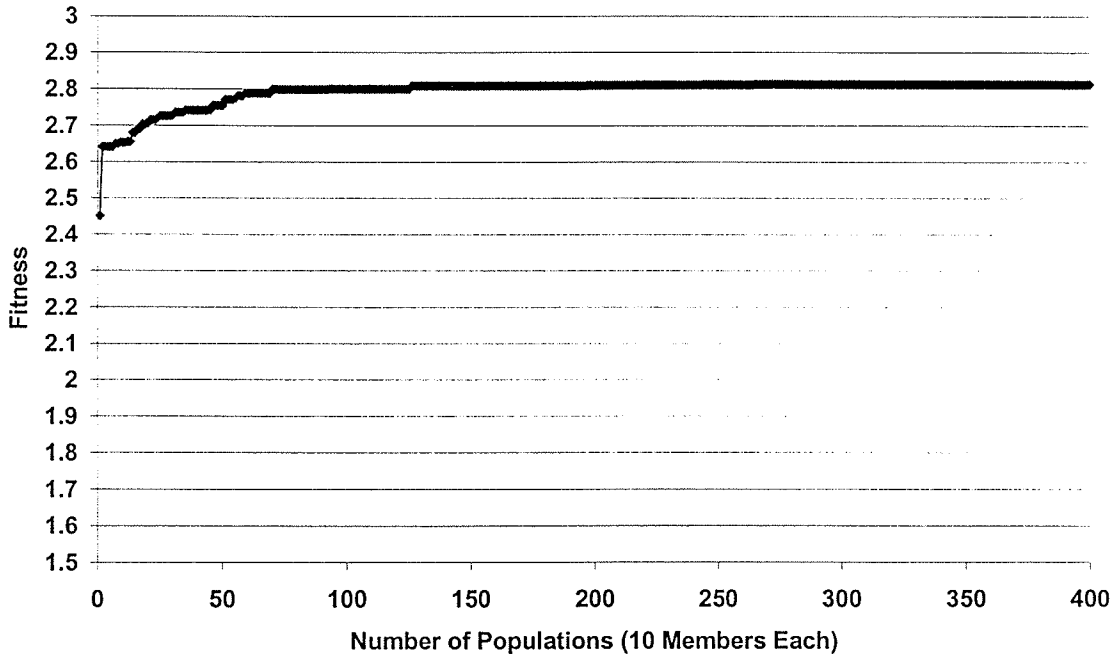

Figure B.24: Plot of Fitness versus Number of Populations for Run 11

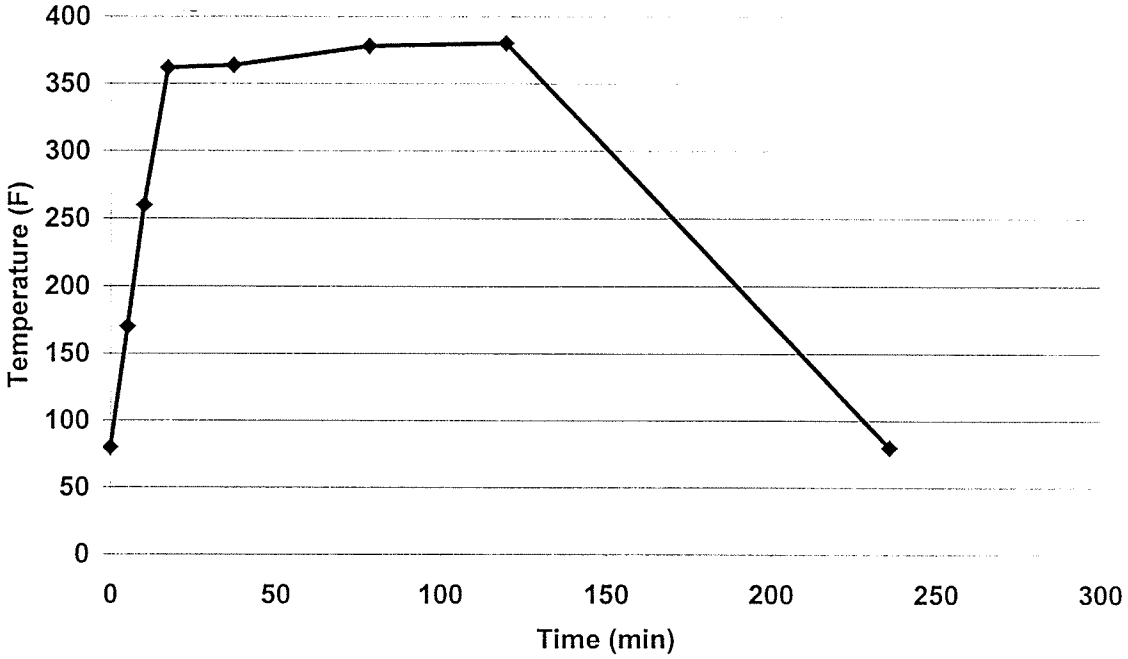

Figure B.25: Plot of Optimal Cycle from Run 12

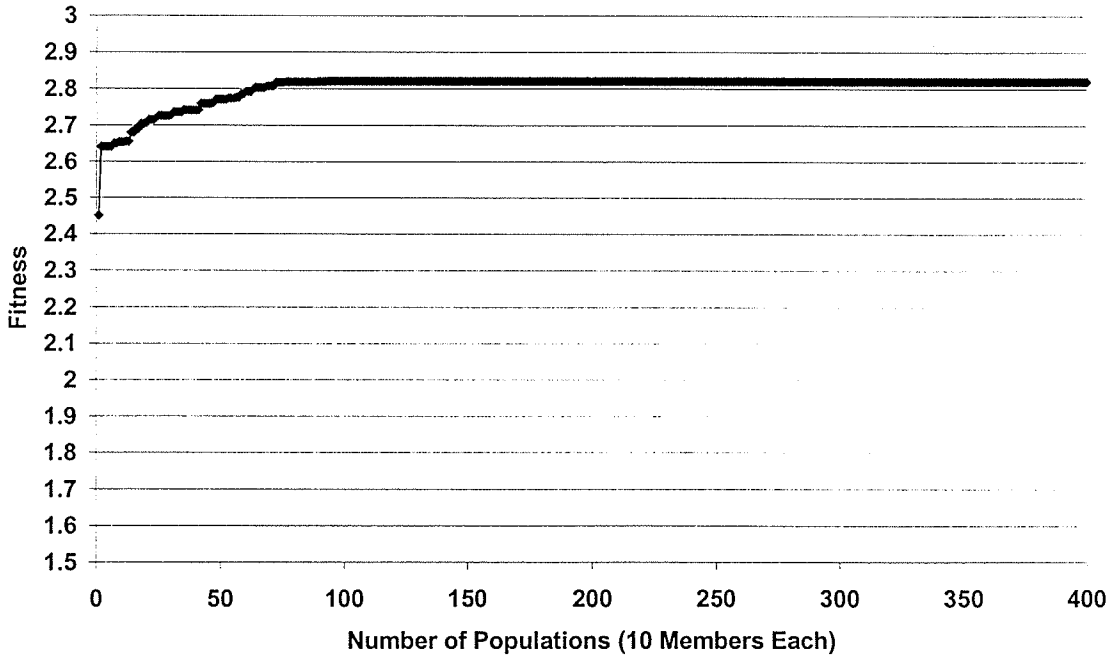

Figure B.26: Plot of Fitness versus Number of Populations for Run 12

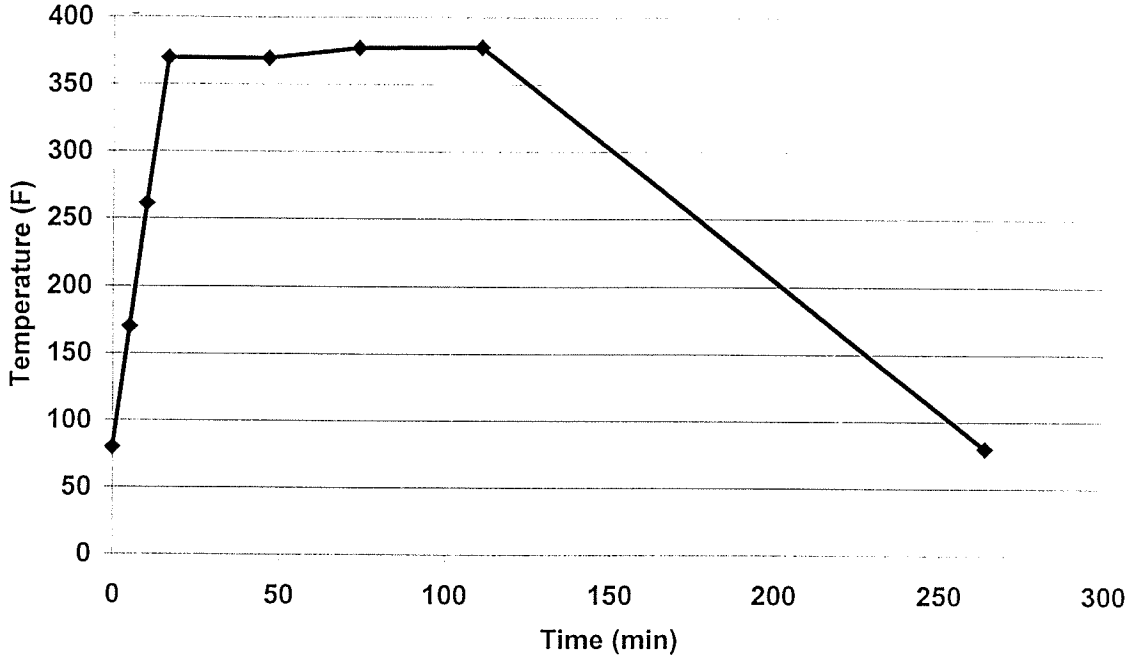

Figure B.27: Plot of Optimal Cycle from Run 13

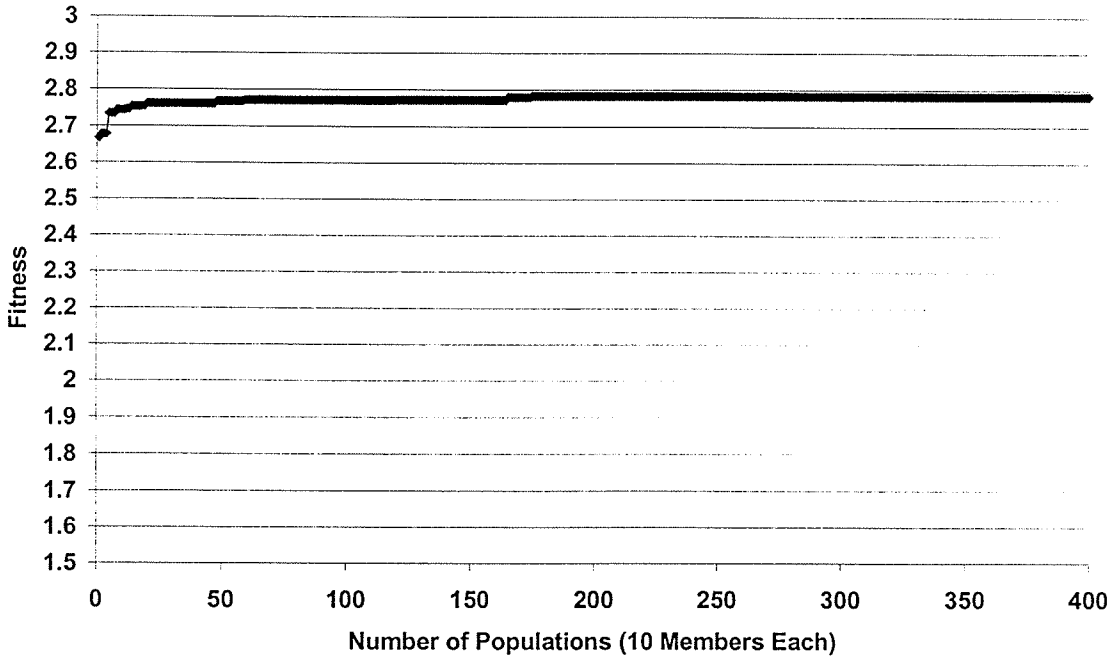

Figure B.28: Plot of Fitness versus Number of Populations for Run 13

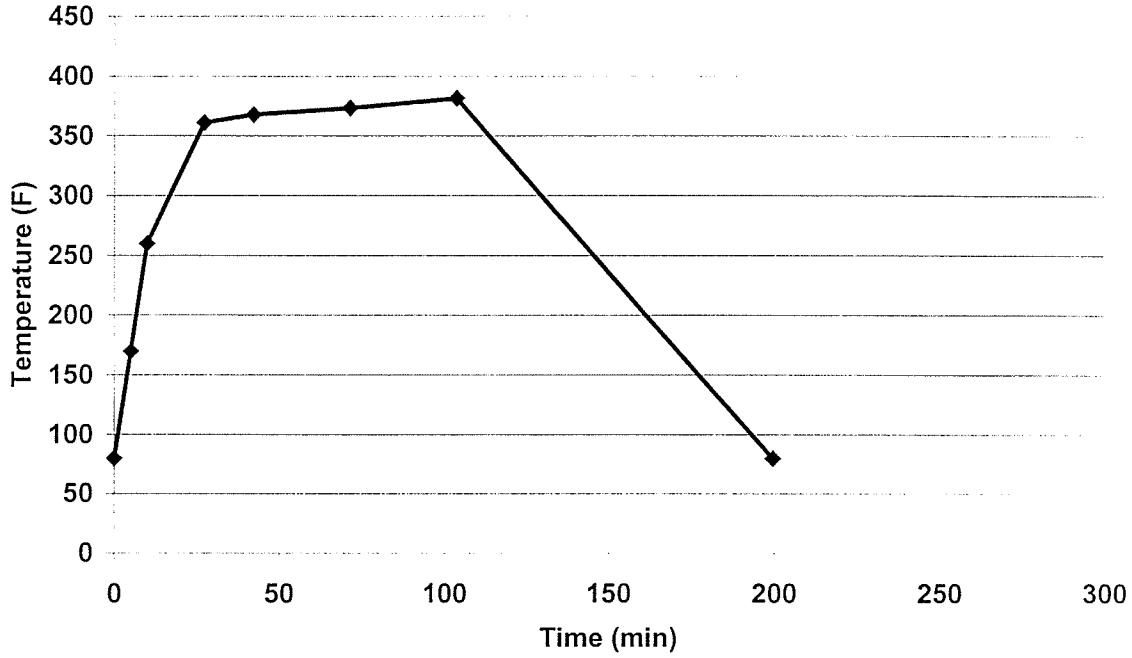

Figure B.29: Plot of Optimal Cycle from Run 14

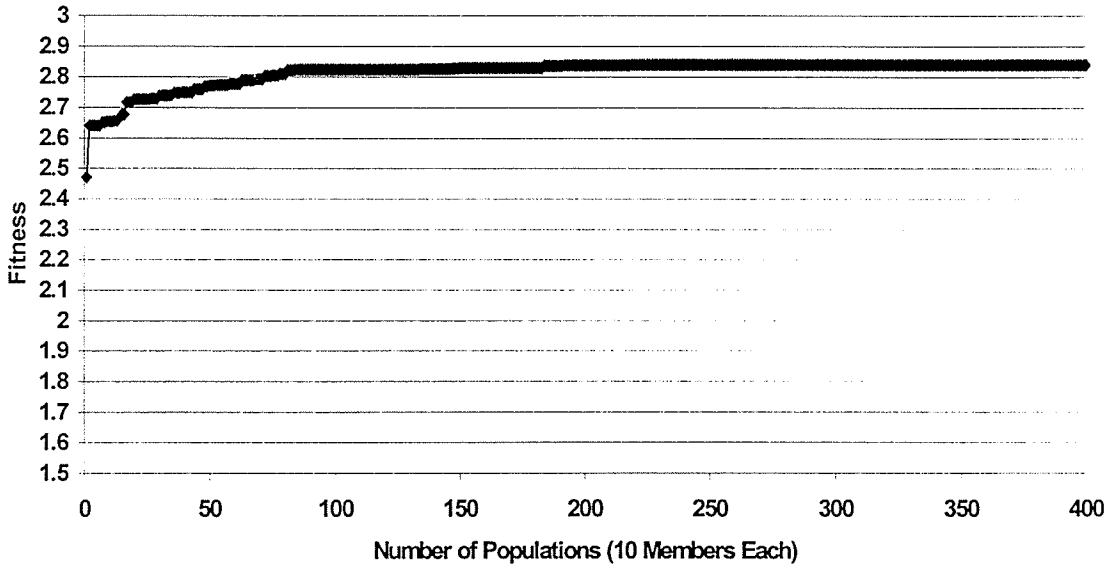

Figure B.30: Plot of Fitness versus Number of Populations for Run 14

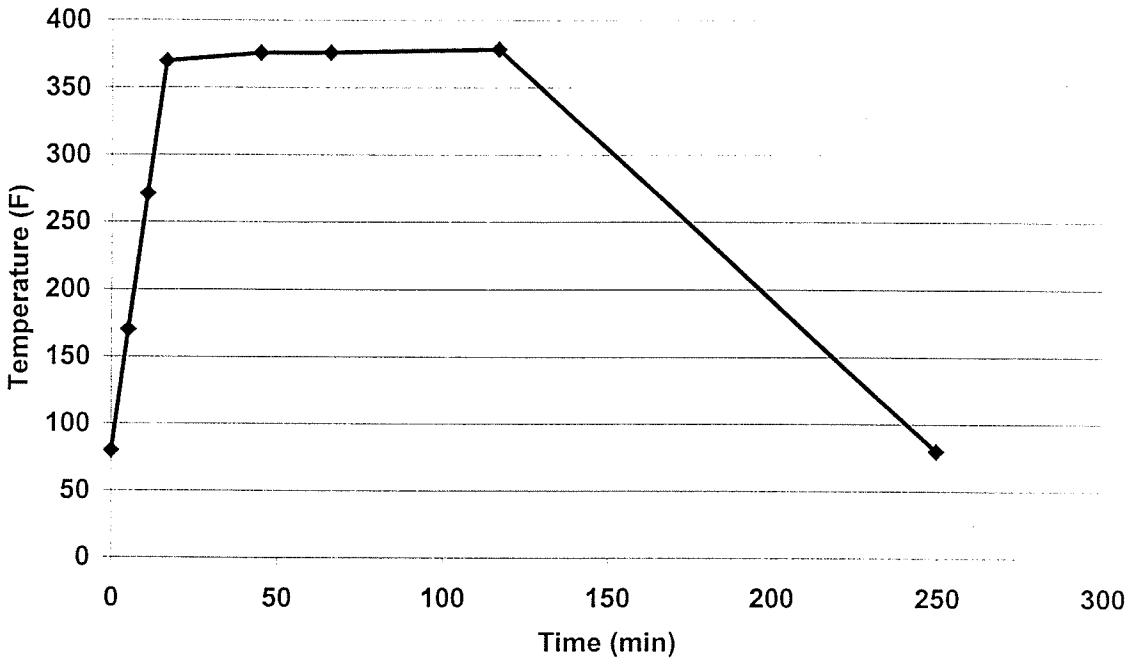

Figure B.31: Plot of Optimal Cycle from Run 15

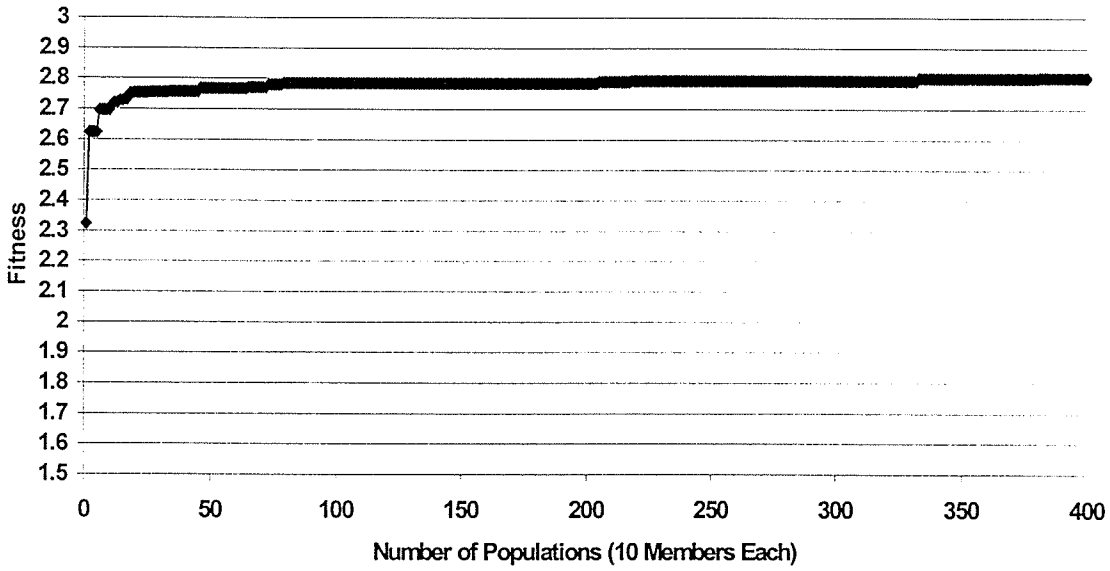

Figure B.32: Plot of Fitness versus Number of Populations for Run 15

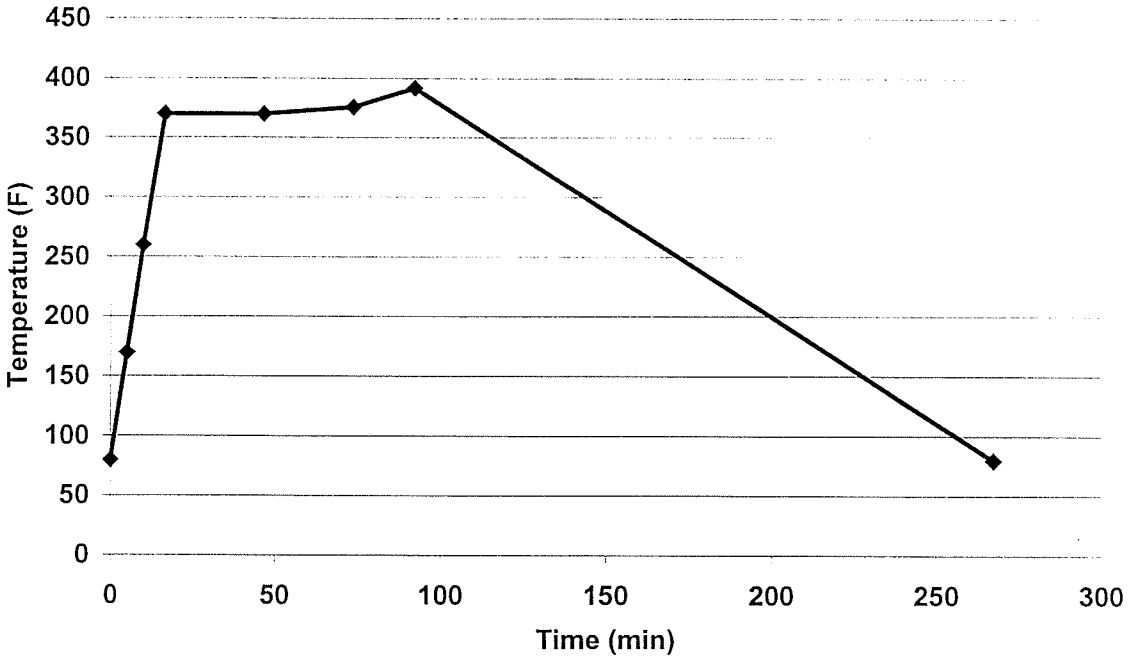

Figure B.33: Plot of Optimal Cycle from Run 16

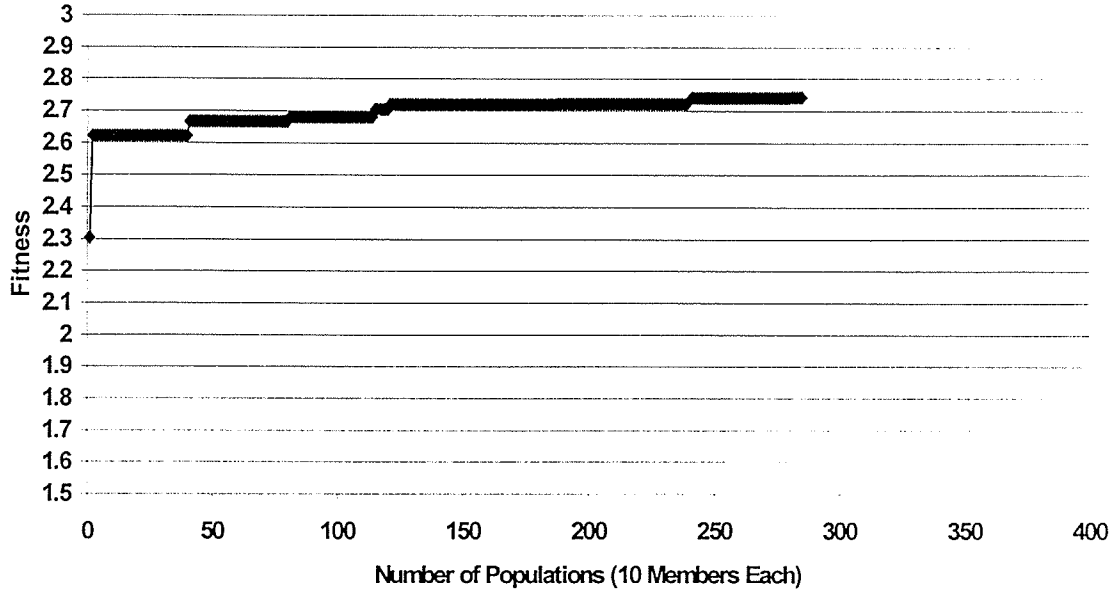

Figure B.34: Plot of Fitness versus Number of Populations for Run 16

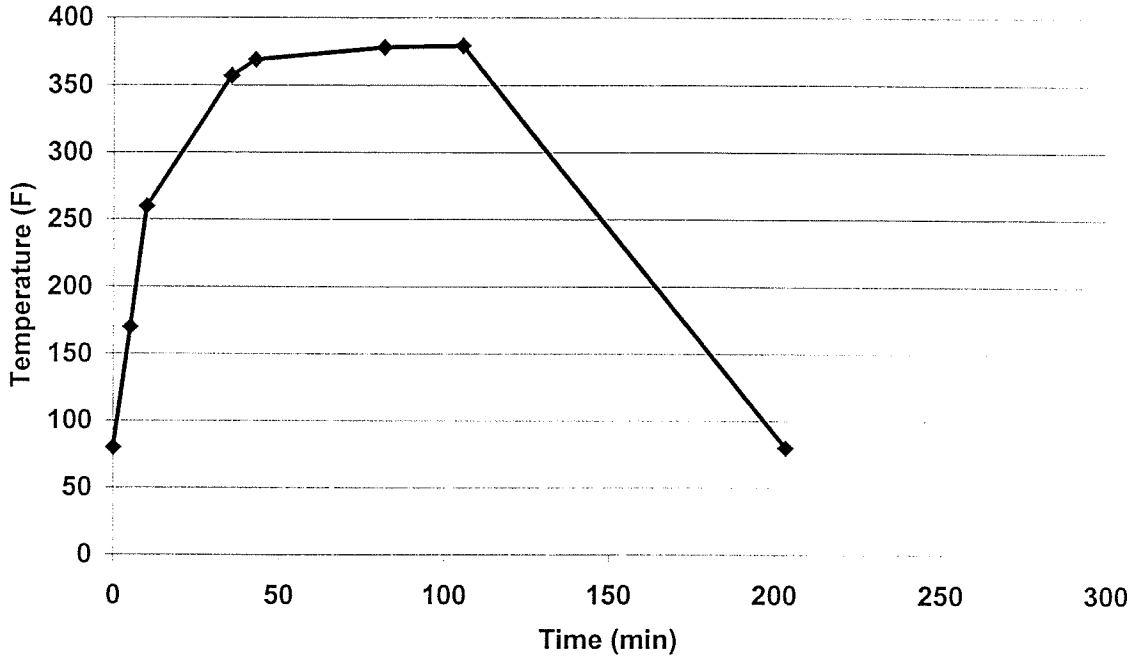

Figure B.35: Plot of Optimal Cycle from Run 17

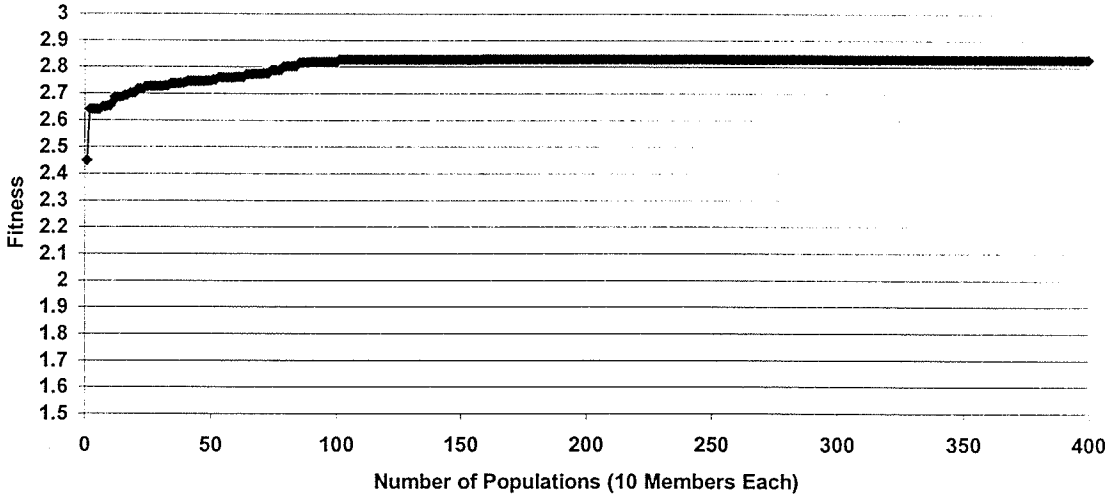

Figure B.36: Plot of Fitness versus Number of Populations for Run 17

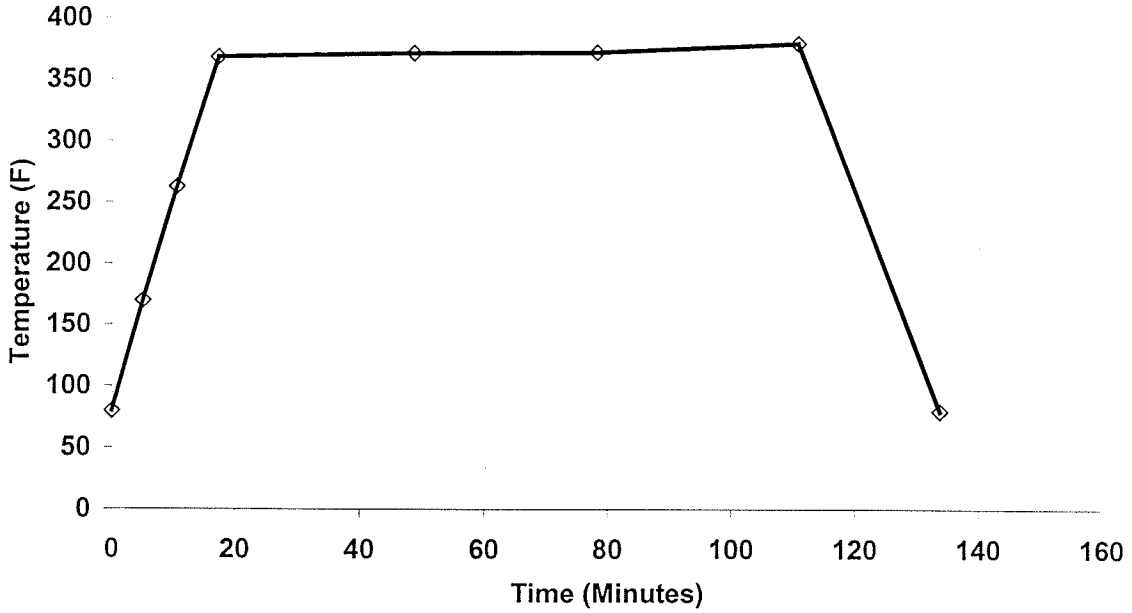

Figure B.37: Plot of Optimal Cycle from Run 18

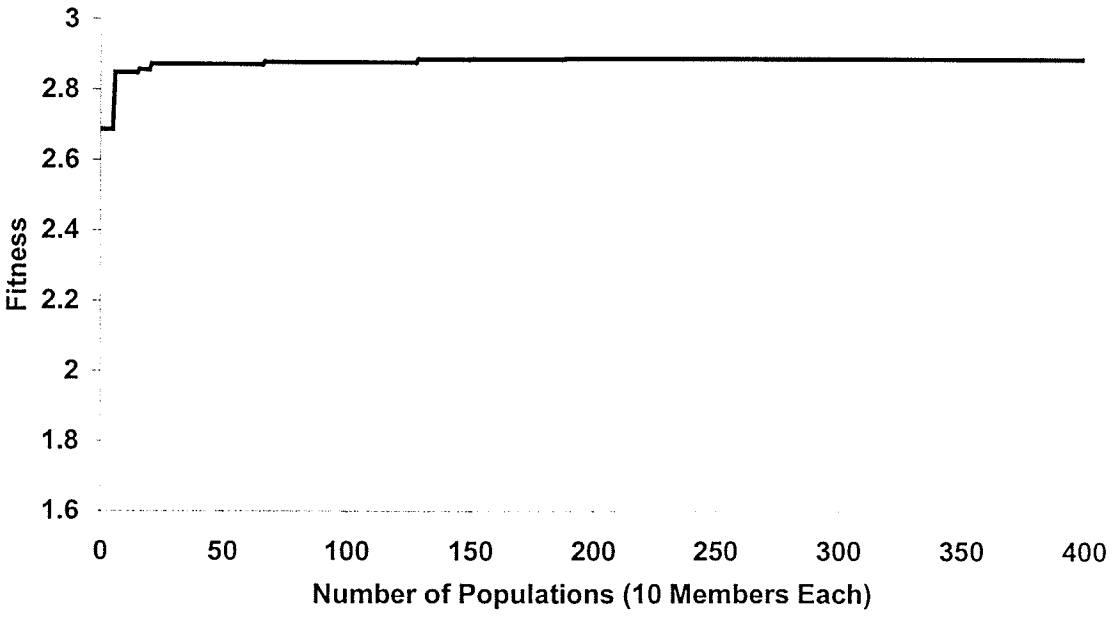

Figure B.38: Plot of Fitness versus Number of Populations for Run 18