GENERALIZED REDUCED GRADIENT ALGORITHM FOR A LOCAL OPTIMUM OF A COST FUNCTION BUILT THROUGH SIMULATION

by

Roberto L. Seijo Vidal

A thesis submitted in partial fulfillment
of the requirements for a degree of

MASTER OF SCIENCE

in

Industrial Engineering

University of Puerto Rico
Mayagüez Campus
2004

Approved by:

María Irizarry, Ph.D. — 8/12/04
Member, Graduate Committee — Date

Omell Pagán Parés, Ph.D. — 8/18/04
Member, Graduate Committee — Date

Sonia M. Bartolomei-Suárez, Ph.D. — August 12, 2004
President, Graduate Committee — Date

Freya Toledo, M.S. — 8-12-04
Representative of Graduate Studies — Date

Agustín Rullán Toro, Ph.D. — 8-12-04
Chairperson of the Department — Date

ABSTRACT

This project is the continuation of a previous simulation study based on a trial and error approach that pretended to find a better system. This new phase pursued a scientific approach for the simulation study in order to identify the best alternative: sensitivity analysis, design of experiments, regression analysis for metamodeling purposes, and optimization. Typical simulation optimization methods were not of practical value for this application. An optimization tool based on mathematical programming was developed using Microsoft's Excel Solver. The tool was validated in terms of the metamodels accuracy and the capacity to find a local optimum within the search region. It was concluded that additional experimental designs were needed in order to find the global optimum. Nevertheless, the tool was valid for the practical application of this project. Finally, it was also concluded that the scientific approach rendered better results than the trial and error approach.

RESUMEN

Este proyecto es la continuación de un estudio de simulación que pretendía resolver un problema de aplicación práctica siguiendo una metodología de prueba y error. Esta nueva fase perseguía encontrar la mejor alternativa siguiendo una metodología científica: análisis de sensitividad, diseño de experimentos, metamodelos y optimización. Se determinó que los métodos de optimización comúnmente utilizados en simulación no eran prácticos para este problema. Se desarrolló una herramienta de optimización basada en programación matemática con el uso de "Microsoft Excel Solver". La herramienta fue validada en términos de la precisión de los metamodelos y la capacidad para encontrar el óptimo dentro de la región de búsqueda. Se concluyó que para determinar el óptimo verdadero, era necesario realizar experimentos adicionales. Sin embargo, se demostró que la herramienta era válida para el problema que se pretendía resolver. Por último, se demostró que el método científico rindió mejores beneficios que el método de prueba y error.

DEDICATORY

This project is dedicated to my wife and children, Mayra, Susan, Robertito and Daniel, for being my most precious treasure and for reminding me every day about the blessings that God has given me in life.

It is also dedicated to the memory of my mother and to my father; for teaching me the values and principles that rule my adult life.

The final dedication goes to my sisters and brother, nephews and nieces, and my family in law; for teaching me the importance of family life.

ACKNOWLEDGEMENTS

I want to express my most sincere gratitude to the president of the graduate committee, Dr. Sonia Bartolomei-Suárez for her support, guidance and patience. Her contribution was of great value for the execution and completion of this project.

I wish to thank you the members of the graduate committee, Dr. María Irizarry and Dr. Omell Pagán Parés, and the representative of the graduate studies, Freya Toledo, M.S. for their support, critical lecture and corrections to the written document. A special thank you goes to Dr. David González and Dr. Noel Artiles, for their wise advices and encouragement. I appreciate the Industrial Engineering Department at the University of Puerto Rico, Mayagüez Campus, for the opportunity given to complete the master and financial support provided.

My special gratitude goes to my wife, Mayra Méndez, M.S., for the support given with the edition of this document; but most important, for her encouragement, friendship, companionship, and unconditional love.

Finally, my gratitude goes to several of my peers as graduate students: to Julian Rivera for his early contributions to the data gathering, development, verification and validation of the simulation model, even though those stages are not part of the scope of this project, they are the foundation for the simulation output analysis; Jannette Pérez for her support with the development of the optimization tool; Dennis Rosario and Geovanie Galán for their time availability during several informal and improvised brainstorming sessions. Interchanging ideas among graduate students is an always learning process.

TABLE OF CONTENTS

CHAPTER I. Introduction……1

1.1. Introduction……..1

1.2. Justification……..1

1.3. Description of the System……3

1.4. Definition of the Problem……6

1.5. Objective and Scope……..10

1.6. Summary……12

CHAPTER II. Literature Review……...13

2.1. Introduction……13

2.2. Simulation Process……13

2.3. Output Data Analysis……16

2.4. Comparison of Alternative System Designs……..27

2.5. Experimental Design and Optimization……30

2.6. Simulation Optimization……42

2.7. Solver Methodology……...46

2.8. Summary……48

CHAPTER III. Methodology……..50

3.1. Introduction……50

3.2. Sensitivity Analysis……...51

3.3. Reducing the Problem with Experimental Designs……...64

3.4. Development of Metamodels……71

3.5. Optimization Tool……..77

3.6. Summary……84

CHAPTER IV. Analysis and Results.....86

4.1. Introduction……..…86

4.2. Sensitivity Analysis……....86

4.3. Reducing the Problem with Experimental Designs……..123

4.4. Development of Metamodels……....132

4.5. Optimization Tool……..…181

4.6. Comparison of Methodologies…….185

4.7. Summary……...188

CHAPTER V. Conclusions and Recommendations……...190

5.1. Conclusions……..….190

5.2. Recommendations……..…194

Bibliography……...196

LIST OF TABLES

Table 1.1 Data and input parameters from system……..6

Table 1.2 Cost related data……..8

Table 1.3 Maximum machine up-time and personnel utilization.....8

Table 2.1 Theoretical values for t-paired test……29

Table 2.2 Design matrix for a 2^k factorial design……..33

Table 3.1 Input Parameters – "Actual System" vs. "3000 Units Through Test"……...55

Table 3.2 Input Parameters – "New System at 3000 Units Through Test" vs. "6 Testing Machines"…….57

Table 3.3 Input Parameters – "New System at 3000 Units Through Test" vs. "90% Yield"……...58

Table 3.4 Input Parameters – "New System at 3000 Units Through Test" vs. "30% Reduction Test. Proc. Time"……....59

Table 3.5 Input Parameters – "New System at 3000 Units Through Draw Out" vs. "4 Assemblers"……..60

Table 3.6 Input Parameters – "New System at 3000 Units Through Draw Out" vs. "4 Final Inspectors"……...61

Table 3.7 Input Parameters – "New System at 3000 Units Through Draw Out" vs. "3 Draw Out Machines"……62

Table 3.8 Input Parameters – "New System at 3000 Units Through Packing" vs. "3 Packers"……...63

Table 3.9 2^{8-2} fractional factorial design with center point – Factor levels……...65

Table 3.10 2^3 factorial design - Factor levels……....69

Table 3.11 2^3 central composite factorial design - Factor levels……...72

Table 3.12 2^3 central composite factorial design - Factor levels……...74

Table 3.13 2^3 central composite factorial design - Factor levels……..……..…..76

Table 3.14 Validation Scenarios for the metamodels accuracy……..…82

Table 3.15 Validation Scenarios for the experimental regions…….....83

Table 4.1 Validation of the warm-up period – "Actual System" model……...….89

Table 4.2 Validation of the warm-up period – "Warm-up Period Validation" model…….....90

Table 4.3 Validation of the warm-up period – Hypothesis tests……...…….....91

Table 4.4 "3000 Units Through Test" – required number of runs…….....93

Table 4.5 "Actual System" vs. "3000 Units Through Test" – Hypothesis test……...…95

Table 4.6 "New System 3000 Units Through Test" – required number of replicates……...…98

Table 4.7 "6 Testing Machine" – required number of runs……...99

Table 4.8 "New System 3000 Units Through Test" vs. "6 Testing Machines" – Hypothesis test……..100

Table 4.9 "90% Yield" – required number of runs……..….102

Table 4.10 "New System 3000 Units Through Test" vs. "90% Yield" – Hypothesis test…….....103

Table 4.11 "30% Reduction Testing Process Time" – required number of runs……...105

Table 4.12 "New System 3000 Units Through Test" vs. "30% Red. Test Proc. Time" – Hypothesis test……...…….....106

Table 4.13 "New System 3000 Units through Draw Out" – required number of runs…….…...……..…109

Table 4.14 "4 Assemblers" – required number of runs.....110

Table 4.15 "New System 3000 Units Through Draw Out" vs. "4 Assemblers" – Hypothesis test…….……...…….....111

Table 4.16 “4 Final Inspectors” – required number of runs……113

Table 4.17 “New System 3000 Units Through Draw Out” vs.
“4 final inspectors” – Hypothesis test……....114

Table 4.18 “3 Draw Out Machines” – required number of runs…….....116

Table 4.19 “New System 3000 Units Through Draw Out” vs.
“3 Draw Out Machines” – Hypothesis test……117

Table 4.20 “New System 3000 Units Through Packing” – required
number of runs…….…..120

Table 4.21 “3 Packers” – required number of runs…….….121

Table 4.22 “New System 3000 Units Through Packing” vs.
“3 Packers” – Hypothesis test……122

Table 4.23 Data sheet for 2^3 factorial design……...128

Table 4.24 Data sheet for 2^3 central composite factorial design
– 3 testing machines……...134

Table 4.25 Data sheet for 2^3 central composite factorial design
– 4 testing machines……...149

Table 4.26 Data sheet for center points 2^3 central composite
factorial design – 4 testing machines……152

Table 4.27 Data sheet for 2^3 central composite factorial design
– 5 testing machines……...166

Table 4.28 Optimization tool and simulation results for validation purposes……183

Table 4.29 Accuracy (error %) of the metamodel vs. the simulation……..183

Table 4.30 Scenario #1 – Experimental region……184

Table 4.31 Scenario #2 – Experimental region……184

Table 4.32 Comparison actual vs. previous project……186

Table 4.33 Buffer requirements……...188

LIST OF FIGURES

Figure 1.1 Conceptual model for the system under study……...4

Figure 2.1 Transient and steady-state density functions……21

Figure 2.2 Meta-heuristic approach for simulation optimization……..45

Figure 3.1 Cube Plot of the 2^{8-2} fractional factorial design with center points……..66

Figure 3.2 Cube Plot of the 2^3 factorial design……..69

Figure 3.3 Data entry screen……..79

Figure 3.4 Optimization Screen……79

Figure 4.1 Normality test – "Actual System" model……87

Figure 4.2 Normality test – "Warm-up Period Validation" model……88

Figure 4.3 Normality test – "3000 Units Through Test" model……92

Figure 4.4 Normality test – "New System 3000 Units Through Test" model…….....96

Figure 4.5 Normality test – "6 Testing Machines" model. ……...97

Figure 4.6 Normality test – "90% Yield" model……101

Figure 4.7 Normality test – "30% Reduction Testing Process Time" model……...104

Figure 4.8 Normality test – "New System at 3000 Units Through Draw Out" model……107

Figure 4.9 Normality test – "4 Assemblers" model……108

Figure 4.10 Normality test – "4 Final Inspectors" model……112

Figure 4.11 Normality test – "3 Draw Out Machines" model……115

Figure 4.12 Normality test – "New System 3000 Units Through Packing" model……118

Figure 4.13 Normality test – "3 Packers" model……..……119

Figure 4.14 Main effects plot for 2^{8-2} fractional factorial design with center points……..125

Figure 4.15 Box plot for the line balancing issue - 2^{8-2} fractional factorial design with center points……125

Figure 4.16 Box plot for different factors interaction - 2^{8-2} fractional factorial design with center points……126

Figure 4.17 Box plot for up-time draw out machine - 2^{8-2} fractional factorial design with center points……127

Figure 4.18 Main effects plot - 2^{3} factorial design……..129

Figure 4.19 Interaction plot - 2^{3} factorial design…….130

Figure 4.20 Boxplot labor cost - 2^{3} factorial design……131

Figure 4.21 Boxplot % utilization final inspector - 2^{3} factorial design ……..131

Figure 4.22 Boxplot %up-time draw out machine - 2^{3} factorial design……..132

Figure 4.23 Regression analysis for throughput – 3 testing machines……135

Figure 4.24 Residual plots for throughput – 3 testing machines…….137

Figure 4.25 Normal probability plot for throughput – 3 testing machines……..137

Figure 4.26 Runs test for throughput – 3 testing machines…….138

Figure 4.27 Surface plot for throughput vs. inter-arrival time and yield – 3 testing machines…….139

Figure 4.28 Regression analysis for Uptime testing machine – 3 testing machines……..140

Figure 4.29 Residual plots for Uptime testing machine – 3 testing machines……..141

Figure 4.30 Normal probability plot for Uptime testing machine – 3 testing machines……..141

Figure 4.31 Runs test for Uptime testing machine – 3 testing machines…….142

Figure 4.32 Surface plot for Uptime testing machine vs. inter-arrival time and testing process time – 3 testing machines……..143

Figure 4.33 Regression analysis for WIP – 3 testing machines……...144

Figure 4.34 Residual plots for WIP – 3 testing machines……145

Figure 4.35 Normal probability plot for WIP – 3 testing machines……145

Figure 4.36 Runs test for WIP – 3 testing machines……146

Figure 4.37 Surface plot for WIP vs. yield and testing process time – 3 testing machines……..146

Figure 4.38 Surface plot for WIP vs. inter-arrival time and yield – 3 testing machines……..147

Figure 4.39 Surface plot for WIP vs. inter-arrival time and testing process time – 3 testing machines…….147

Figure 4.40 Regression analysis for throughput – 4 testing machines……150

Figure 4.41 Residual plots for throughput – 4 testing machines…….151

Figure 4.42 Normal probability plot for throughput – 4 testing machines……..153

Figure 4.43 Runs test for throughput – 4 testing machines…….153

Figure 4.44 Surface plot for throughput vs. inter-arrival time and yield – 4 testing machines……..154

Figure 4.45 Regression analysis for Uptime testing machine – 4 testing machines……..155

Figure 4.46 Residual plots for Uptime testing machine – 4 testing machines……..156

Figure 4.47 Normal probability plot for Uptime testing machine – 4 testing machines……..157

Figure 4.48 Runs test for Uptime testing machine – 4 testing machines…….157

Figure 4.49 Surface plot for Uptime testing machine vs. inter-arrival time and testing process time – 4 testing machines……..158

Figure 4.50 Regression analysis for WIP – 4 testing machines……...159

Figure 4.51 Residual plots for WIP – 4 testing machines……159

Figure 4.52 Normal probability plot for WIP – 4 testing machines……160

Figure 4.53 Regression analysis for WIP – 4 testing machines……..160

Figure 4.54 Residual plots for WIP – 4 testing machines……...162

Figure 4.55 Normal probability plot for WIP – 4 testing machines……162

Figure 4.56 Runs test for WIP – 4 testing machines……163

Figure 4.57 Surface plot for WIP vs. yield and testing process time – 4 testing machines……..164

Figure 4.58 Surface plot for WIP vs. inter-arrival time and yield – 4 testing machines……..164

Figure 4.59 Surface plot for WIP vs. inter-arrival time and testing process time – 4 testing machines…….165

Figure 4.60 Regression analysis for throughput – 5 testing machines……167

Figure 4.61 Residual plots for throughput – 5 testing machines…….168

Figure 4.62 Normal probability plot for throughput – 5 testing machines……...169

Figure 4.63 Runs test for throughput – 5 testing machines…….170

Figure 4.64 Surface plot for throughput vs. inter-arrival time and yield – 5 testing machines…….171

Figure 4.65 Surface plot for throughput vs. inter-arrival time and yield – 5 testing machines……..171

Figure 4.66 Regression analysis for utilization packing personnel – 5 testing machines……..172

Figure 4.67 Residual plots for utilization of packing personnel – 5 testing machines…….173

Figure 4.68 Normal probability plot for utilization packing personnel – 5 testing machines……174

Figure 4.69 Runs test for utilization packing personnel – 5 testing machines……..174

Figure 4.70 Surface plot for utilization packing personnel vs. inter-arrival time and yield – 5 testing machines…….175

Figure 4.71 Surface plot for utilization packing personnel vs. yield and testing process time – 5 testing machines…….176

Figure 4.72 Surface plot for utilization packing personnel vs. inter-arrival time and testing process time – 5 testing machines…….176

Figure 4.73 Regression analysis for WIP – 5 testing machines……...177

Figure 4.74 Residual plots for WIP – 5 testing machines……178

Figure 4.75 Normal probability plot for WIP – 5 testing machines……179

Figure 4.76 Runs test for WIP – 5 testing machines……179

Figure 4.77 Surface plot for WIP vs. yield and testing process time – 5 testing machines……..180

Figure 4.78 Surface plot for WIP vs. inter-arrival time and yield – 5 testing machines……..180

Figure 4.79 Surface plot for WIP vs. inter-arrival time and testing process time – 5 testing machines…….181

Figure 4.80 Probability curve scenario #2 – Throughput (90% yield; 70% testing process time)…….187

LIST OF APPENDIXES

APPENDIX A. Summary Previous Project……199

APPENDIX B. SIMAN Codes…..……...205

APPENDIX C. Verification warm-up period – Input parameters……...214

APPENDIX D. 2^{8-2} Fractional Factorial Design with Center Points…….……..215

APPENDIX E. 2^3 Factorial Design……..……...218

APPENDIX F. Central Composite Design……..……219

APPENDIX G. Visual Basic Program-Optimization Tool……..……221

CHAPTER I. Introduction

1.1. Introduction

This project is based on a real scenario in which simulation modeling is used in order to understand the behavior of the system. Design of experiments and optimization are key elements of the simulation output analysis in order to identify critical parameters and their relationships to multiple responses or output variables. Understanding this relationship allows to build mathematical expressions for the output variables which is the foundation for the optimization.

The project is the continuation of a previous project that was developed as part of the requirements for a computer simulation course towards the Master in Engineering degree. Contrary to the previous project which was a practical application with direct involvement of members of the company including management, this new phase is more of an academic interest with no interrelation with the company representatives. In the previous phase, focus was given to the data gathering and analysis, to the development, verification and validation of the model, and to the experimentation by means of a trial and error approach. In this new phase, focus will be given to the output analysis by means of sensitivity analysis, design of experiments, metamodeling and optimization.

1.2. Justification

Computer simulation is a technique widely used in operations research and management science. It can be defined as the process in which a real or a proposed system[1] is mimic or imitated by designing a model[2] using the computer. The purpose of simulation is to understand the behavior of the system over time in order to evaluate different improvement alternatives or strategies for its operation. This is usually accomplished by experimenting with the simulation model for a given set of conditions.

[1] System – the facility or process of interest, either actual or planned, which is composed of elements that interrelate to achieve a specific objective.

[2] Model - the representation of the system in a set of assumptions that usually take the form of mathematical or logical relationships in order to understand the behavior of the system itself.

Computer simulation includes both the construction of the model and the experimental use of the model.

The process of developing or constructing a model takes the form of different tasks: working with the end users of the simulation results, defining a problem, understanding the system, collecting and analyzing the data, formulating the conceptual model, coding and verifying the simulation model, and then validating its behavior. Usually, a great deal of effort is spent on the development phase of the model instead of on the experimental and output analysis phase. The lack of effort to analyze appropriately the outcome data of the simulation model leads to erroneous inferences about the system under study.

The experimental and output analysis phase is generally less time consuming than the development phase of the model. Output analysis allows the analyst and the end user to understand better the performance of the model leading to better decisions in terms of possible improvements for the system. This project focuses on the output analysis of the simulation model for a real system, in which understanding the relationships between the input parameters and multiple interrelated performance metrics or output variables with restrictions is key to achieve the expected goal of the system. Several input parameters are factors in which the end user can specify any value within a given range of interest. The fact that there are variable input parameters and multiple responses with restrictions, suggests the necessity for using computer simulation with design of experiments[3], metamodels and optimization[4], emphasizing the importance of the output analysis and experimentation of the simulation model.

The previous simulation model was verified and validated and the solution to the problem was attempted using a trial and error approach and by fixing the input parameters to the optimum scenario. Even though the output data was statistically

[3] Design of experiments – a test or series of tests in which purposeful changes are made to the input variables of a process or system so that we may observe and identify the reasons for changes that may be observed in the output response [as per Montgomery (2001)].

[4] Mathematical modeling and optimization – a process in which a decision problem is mathematically modeled allowing the systematic evaluation of all the alternatives or possible solutions of the problem. It has three basic components: decision options or alternatives, problem restrictions and objective criteria.

analyzed in an appropriate manner, the trial and error approach does not guarantee that the best alternative was selected. More than that, the solution provided satisfies only the scenario or condition, in which the input parameters are only at the optimum condition, neglecting the end user to understand the behavior of the system under other conditions. This is indeed the justification for this project since the expectation is to provide the end user with an understanding of the behavior of the system within the given spectrum of the input parameters. Traditional simulation optimization methods are not of practical value for this problem due to the lack of resources availability from the end user and the uncertainty related to the values that the variable input parameters can assume. The approach followed is to express the relation between the simulation input parameters and the responses in terms of a mathematical optimization problem by use of metamodels. Even though the simulation literature recommends caution from the analyst side when pretending to solve simulation optimization problems with deterministic methods, mainly due to the inheriting variability of any simulation model, it is believed that this approach will provide the end user with the full understanding of the system without having to invest in expensive simulation software. Microsoft's Excel Solver is used for the development of the optimization tool, showing to be of practical value due to its universality in today's business world and its use of well known algorithms for solving optimization problems. In addition, we will attempt to demonstrate that the trial and error approach did not reach the optimum solution.

1.3. Description of the System

The previous project took place in a multinational company that manufactures and services electromechanical components used in electrical substations. The mission of the company is to provide innovative products and services of impeccable quality that will add value to the customers, employees and shareholders. Until several years ago, when they introduced a global product, their products were designed for the domestic or international markets. Due to the acceptance that the new product had in both markets, the management identified the necessity for increasing the volume.

The system under study was the assembly line for the new product which is a one-piece flow progressive assembly line. Each product is individually assembled, tested, and final inspected and packed. The testing area was identified by the management as the bottleneck and the area of major concern due to the high cost of the testing equipment, the high rejection rate and the excessive time to test a unit. Management understood that assembly capacity was available or relatively inexpensive to increase, contrary to the testing and final inspection areas. The study was then focused on those two areas as well as packaging.

The purpose of Figure 1.1 is to provide a visual aid for a better understanding of the system. It also shows the total number of employees working at each area.

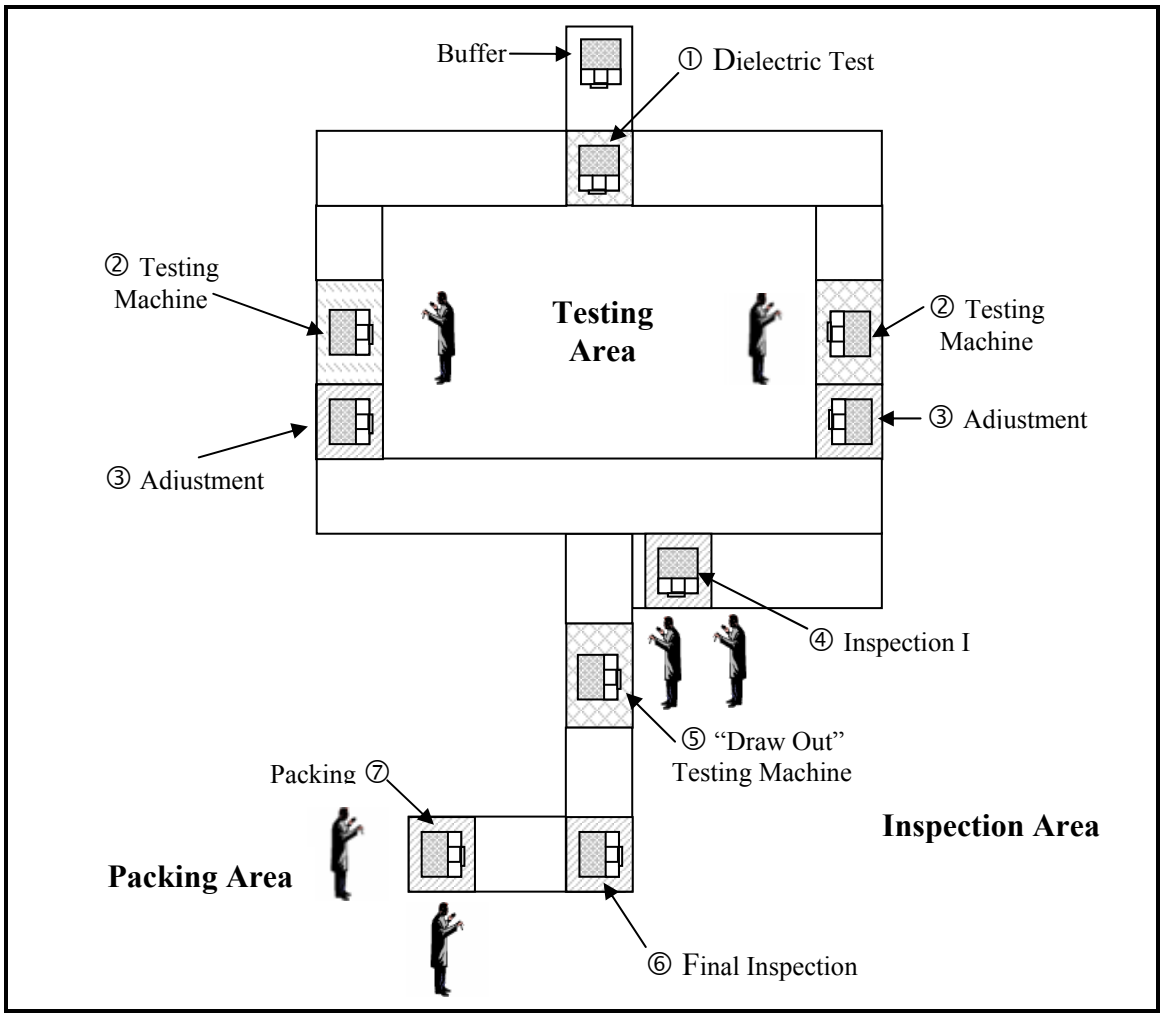

Figure 1.1 Conceptual model for the system under study

The product gets to the testing area from the assembly stations by mean of a manual conveyor. The first working station in the testing area is to perform the dielectric test and there is a buffer before this station. Once the testing person is available, he/she pulls the product out from the buffer to the testing area and performs the dielectric test which is a manual test. The product will be returned to the assembly area if it fails the test. Once the test is completed, the testing person pushes the product to the testing machine that he/she serves and loads the testing machine if available. Otherwise, the product is buffered until the machine is available. There are two identical testing machines served by independent testing personnel. Once the testing machine is loaded, the testing person is available for other responsibilities since the machine runs automatically. After completion of the testing process, the testing person unloads the testing machine and returns the product to the assembly area if the product was rejected. Otherwise, he/she proceeds to perform a visual inspection and some required adjustments. Once completed, the testing person pushes the product to the assembly area if the product was rejected or to the final inspection area and it is buffered until the assembler or final inspector are available. Task priority in the testing area is given first to the loading/unloading of the testing machine, second to the dielectric test and finally to the adjustment process.

Personnel classified as an assembler or the final inspector, whoever is available but priority given to the assembler performs a visual inspection to the product and takes it to the assembly area if it is rejected or proceeds to load the draw out machine if the product is accepted. The draw out test is then performed and it requires the presence of the personnel all the time. Once again, priority is given to the assembler for this test. If rejected, the product will be taken to the assembly area; otherwise it will be buffered for the final inspection which can only be done by the final inspector. Final inspection is performed and once again, the product will be routed to the assembly area if rejected or pushed into the packing buffer. The product will finally be packed by packing personnel and stacked until the shipper proceeds to move it to the dock area. The shipping process was not part of the scope of the project.

Table 1.1 shows the number of employees working at each of the three main areas. It also shows the statistical distribution that follows the process time for each working station and the acceptance levels or yield for the working stations in which the product is inspected or tested. Please note that "N/A" in this table means that no inspection is performed.

Please refer to Appendix A for more information related to the simulation model used for this project. This project will not cover the data gathering and analysis, nor the development, verification and validation of the model. These critical stages of any simulation process were already covered during the previous project as specified in section 1.1. Appendix A will cover the assumptions of the simulation model, and the definition of entities, attributes, type of events and state variables. It will also cover a brief summary of the data gathering and analysis, verification and validation of the model, trial and error runs, conclusions and recommendations.

Table 1.1 Data and input parameters from system

Working Area	Number of Employees	Employee Classification	Process	Process Time (Min.)	Yield (%)
Testing	2	Tester	Dielectric Test	1.06 + Logn (0.423,0.266)	99.0
			Loading Testing Machine	1+Logn (0.334, 0.231)	N/A
			Testing	3 + 52 Beta (2.93, 4.65)	72.6
			Unloading Testing Machine	1.05+Weib (0.17,2.03)	N/A
			Adjustment	1.03+Logn (0.369,0.202)	100.0
Inspection	2	1 Assembler; 1 Inspector	Inspection I	Tria (5.5,10.5,18.5)	97.5
			Draw Out Test	7.5 + Weib(8.6,2.32)	96.1
			Final Inspection	3.5 + Expo (3.27)	94.8
Packing Area	2	Packer	Packing	8 + 9 * Beta (1.15, 0.554)	N/A

1.4. Definition of the Problem

The problem will be defined in two categories: the practical application of computer simulation for the system under study, and the academic interest on simulation output analysis for systems with multiple response variables and restrictions.

1.4.1. Practical Application

As previously mentioned in section 1.3, the system of interest in this project is the assembly line for a new product manufactured by a multinational company, in particular the testing, final inspection and packing areas. The testing area was identified by the management of the company as the bottleneck and the area of major concern due to the high cost of the testing equipment and the excessive time for testing a unit. In addition, there was a high rejection rate in this area and the management was interested in knowing how this poor quality performance was impacting the performance of the line.

Due to the market acceptance for the new product, the management of the company identified the necessity for increasing their volume from 800 units per month to 3000 units per month. At first glance, identifying the requirements for the increment in volume seemed to be a relatively easy task since the new product was under production for about two years. Nevertheless, the local management was restricted to find the best alternative with the minimum capital investment and operational cost due to the new business acquisition strategy of the main company. For the purpose of the analysis, capital investment was related to the acquisition of new equipments and work-in-process inventory, and operational cost was mostly related to labor. Table 1.2 shows the costs associated to labor and equipment acquisition. In order to facilitate the cost analysis related to depreciation of capital equipment, the straight line method was assumed with a 10 year period. Any equipment with a purchasing cost higher than $2,000 was considered as capital equipment, otherwise, it was considered as expense. The estimated cost for a unit in work-in-process is $1500 and each unit is sold by $3500. Please note that “N/A” in this table means that the equipment used in that working area, if any, is an expense item rather than capital equipment, in other words, there is no depreciation associated with that piece of equipment.

Table 1.2 Cost related data

Working Area	Employee Classification	Labor Cost ($/Hr-Emp)	Equipment	Capital Expenditure ($/Equip)	Book Value ($)
Testing	Tester	7.35	Dielectric Testing Machine	300	N/A
			Testing Machine	350,000	50,000
Inspection	Assembler	6.35	Draw Out Testing Machine	50,000	
	Inspector	8.35			10,000
Packing Area	Packer	6.85	Packing	N/A	N/A

The management was facing machine downtime in particular with the testing equipment and the draw out machine. Even though the breakages were infrequent, the management set up a maximum limit for the equipment up-time in order to allow the simulation study to consider machine downtime since no data was available for the stoppages. Similarly, limits were set up for the personnel utilization in order to allow for random tasks that could be assigned by the supervisors. Table 1.3 shows the restrictions for the up-time of the equipment and the utilization of the personnel. Please note that "N/A" in this table means that no equipment is used to perform the operation at that particular station.

Table 1.3 Maximum machine up-time and personnel utilization

Working Area	Employee Classification	Maximum Utilization (%)	Equipment	Maximum Up-time (%)
Testing	Tester	70	Dielectric Testing Machine	100
			Testing Machine	95
Inspection	Assembler	90	Draw Out Testing Machine	95
	Inspector	80		
Packing Area	Packer	85	Packing	N/A

The rejection rate was high in particular at the testing area. In addition, the testing process time was also considered to be excessively high. The management goal was to improve the yield at the testing machine from 72.6% to 90%, and reduce the testing process time to 70% of the actual value. In both cases, the investment was not a restriction and it was considered by management as a most do task for their team.

In summary, the management of the company wanted to determine the arrangement needed in order to increase the volume to the projected levels at the minimum operational and capital expenditure costs. The suggested arrangement in terms of equipment and personnel requirements must be determined considering the restrictions of equipment up-time and personnel utilization.

1.4.2. Academic Interest

Simulation modeling and analysis was identified as the right tool to be used due to the stochastic[5] nature of the system under study, which seems to be modeled relatively easy using simulation software like ARENA/SIMAN. Nevertheless, once the model is coded, the subsequent experiments and output analysis are of a complex nature due to the multiple interrelated response variables and their relationship with the input parameters of the process. This relationship with the input parameters suggests design of experiments as a tool to be used as part of the simulation analysis. In addition, the constraints or restrictions over the response variables combined with the fact that there is a specific objective of achieving 3000 units per month at the lowest cost in terms of capital investment and operational costs, suggest the possibility for using metamodels and mathematical programming for optimization purposes. The interrelated response variables will be integrated into the optimization analysis by means of constraints and the cost (objective) function.

The analysis of simulation models of this complexity is usually simplified by reducing the search region, thus defining a manageable number of feasible scenarios. This is achieved by fixing or setting up input parameters to conditions that will impact positively the main goal of the analysis. For example, by fixing the yield to 90% instead of 72.6% and fixing the testing process time to 70% of the actual value instead of 85%, we can expect more throughput[6] and a reduction of the up-time[7] for the testing machine which allows the expected volume to be reached with the least number of machines, thus

---

[5] Stochastic process – a collection of random variables ordered over time, which is all defined on a common sample space [as per Law et al. (1991)].

[6] Throughput - the maximum rate of production in a multistage production process [as per Martinich (1997)].

[7] Up-time – the percentage of time that the testing machine resource was busy or available.

reducing the capital expenditure cost. Nevertheless, the analysis has no value for the end user if those optimum conditions for the input parameters are not reached in real life.

In the previous attempt for solving the problem described in section 1.4.1, the yield and the testing process time were fixed to the optimum conditions, and the experiments followed a trial and error approach. Not only the capacity for understanding the model was limited but there was no guarantee that the alternative suggested was the best alternative in terms of reducing the capital and operational costs.

1.5. Objectives and Scope

As in section 1.4, the objectives and scope of this project will be defined in terms of the practical application and the academic interest on simulation output analysis.

1.5.1. Practical Application

The purpose of this project is to define the arrangement needed in terms of number of equipment and personnel, to support the objective of increasing the throughput from a total of 800 units per month to 3000 units per month with the minimum investment in terms of capital expenditures and operational costs. The above objective must be achieved considering the following restrictions or constraints:

- Work-in-process must not exceed 22 units in average.
- Equipment up-time (%) must not exceed the limit specified on section 1.4.1.
- Labor utilization (%) must not exceed the limit specified on section 1.4.1.

For the particular scenario in which the yield of the testing process is 90% and the testing process time is 70% of the actual value, the probability for the throughput will be defined as well as the size of the buffers.

Achieving a total of 3000 units per month is the main driver for the project. The end user will not be given the option to set up the system for fewer units. Contrary to the yield and testing process time, understanding the behavior of the system for the whole spectrum of alternatives between 800 to 3000 units is not within the scope of this project.

1.5.2. Academic Interest

The main purpose of this project is to provide the end user with a tool of practical value that allows him/her to understand the behavior of the system for the whole spectrum of alternatives or possibilities as defined by the limits specified for the yield and testing process time. This objective will be achieved by focusing on the experimental and the output analysis phase of the simulation model. The use of design of experiments, metamodels[8], and optimization are of major academic interest for the simulation output analysis. These tools as well as the appropriate statistical analysis for the response or output variables of the simulation model are key ingredients for achieving the main objective of this project.

The optimization will be done by expressing the relationship between the input parameters and the simulation responses in terms of a mathematical programming model, which will then be solved by the appropriate optimization method. Caution is suggested when applying deterministic methods for solving simulation optimization problems due to the variability of the simulation model. Nevertheless, as per Banks et al. (2001) it is understood that this risk is minimized if the appropriate statistical analysis is done during the following stages of the simulation study: output analysis with effort on running sufficient replicates for variability reduction, experimental designs analysis with the respective regression for metamodeling building purposes, and finally the selection of the appropriate optimization method. Microsoft's Excel Solver and Visual Basic for Applications will be used for the development of the optimization tool mainly for the following reasons: (1) Microsoft's universality in today's business world makes the optimization tool developed of great practical value for the end user, and (2) the use by Excel Solver of well known algorithms for solving optimization problems, which for this problem is the Generalized Reduced Gradient method.

Finally, another objective of this project is validating as per simulation literature that the use of scientific tools as part of simulation modeling and analysis render better results than the traditional trial and error approach.

---

[8] Metamodel – is an algebraic model of the simulation model [as per Law et al. (1991)].

1.6. Summary

This chapter described the system under study as the testing, final inspection and packing areas on the assembly line of a new product manufactured by a multinational company. Due to market acceptance, the management identified the need for increasing the throughput at the lowest possible cost, both capital and operational. There are some input parameters that are variable and understanding the behavior of the system for the spectrum of possible values that those parameters can assume is critical for the success of the project. There are also multiple interrelated performance measures with restrictions as defined by the management of the company.

It was also mentioned that this project is the continuation of a previous project that focused on finding a solution by means of a trial and error approach, different to this new phase in which focus will be given to the output analysis by means of sensitivity analysis, design of experiments, metamodeling and optimization.

The problem, and objectives and scope, were presented in terms of the practical application and the academic interest in simulation output analysis for multiple responses. With respect to the application, the main objective is to determine the personnel and equipment required, as well as the space requirements due to the buffers. There is special academic interest in formulating the problem as a mathematical model for optimization purposes, and solving the same by means of the Generalized Reduced Gradient algorithm that Microsoft Excel Solver employs for solving smooth non-linear optimization problems.

The following chapter (II) will discuss the literature review with major focus on the output analysis, comparison of alternatives, experimental designs, and simulation optimization. Chapter III will cover the methodology followed for the execution of the project, Chapter IV will discuss the results and statistical analysis, and Chapter V will present the conclusions and recommendations in terms of future research.

CHAPTER II. Literature Review

2.1. Introduction

This section summarizes the literature reviewed related to computer simulation. It focuses mostly on the output analysis, the experimental design and optimization methods rather than on the coding, the verification and validation of the model. Even though simulation is widely used for decision making in the industry, most of the time the effort is placed on building, verifying and validating the model rather than analyzing the output data appropriately. In addition, in many situations the analyst tends to follow the traditional trial and error approach instead of a scientific approach based on experimental design in order to understand the behavior of the system. Unfortunately, the power of simulation is minimized by these two conditions, therefore increasing the probability for making erroneous inferences about the system and defeating the purpose of the simulation study.

2.2. Simulation Process

There is much in the literature related to simulation and how to build a credible simulation study. The following are some references on this subject: Banks et al. (2001), Law et al. (1991), Law (2003), Pegden et al. (1995), and Sargent (1998).

Researchers and experts in computer simulation agree that any successful simulation study must follow some specific steps. Even though they differ in the number of steps, the essence of the suggested process is the same. In order to facilitate the explanation of the process, this section will present the steps as defined by Banks (2001). Nevertheless, the four references will be used to explain each step.

- *Problem formulation* – Clearly stating the problem and assuring that it is understood by the end user of the simulation study. There are occasions where the problem has to be reformulated as the study progresses.
- *Setting of objectives and overall project plan* – Clearly defining the objective and scope of the simulation study and confirming that simulation is the right

or appropriate tool to use. This is also the step for defining the performance measures that will be used to evaluate different configurations for the system under study. Assuring sufficient personnel, management support, computer hardware and software resources should also be considered as part of the planning stage.

- *Model conceptualization* – Developing a preliminary model either graphically or in pseudo-code to define the components, descriptive variables, and interactions (logic) that constitute the system. The model assumptions, algorithms, and data summaries should be documented. The end user should participate to enhance the quality and increase the confidence of the resulting model.
- *Data collection* – identifying and collecting from the system the relevant data that will be used by the model. This data will be used to specify model parameters and probability distributions. In most situations, the required data elements may change as the complexity of the model changes. This data should include the system performance data necessary for the validation of the model. As in the model conceptualization step, it is a good practice to involve the end user.
- *Model translation* – Programming or coding the model. The analyst or modeler must decide whether to program the model using a general purpose programming language such as FORTRAN, C, C++, or by simulation language such as SIMAN, SLAM II, GPSS/H, or by using special purpose simulation software such as ARENA, AWESIM, WITNESS, etc.
- *Verification of the model* – Confirming that input parameters and logical structure of the model are correctly represented in the computer, so that the model operates the way it is intended.
- *Validation of the model* – Confirming that the model is an accurate representation of the real system. This is achieved through the comparison of the model output versus the output of the real system.

- *Experimental design* – designing an experiment that will generate the desired data. For each system configuration or experimental run of interest, the analyst must decide on simulation run length, length of the warm up period and the number of independent replications. The experiment could follow a trial and error approach or a scientific approach based on design of experiment. It is also highly recommended to perform a sensitivity analysis to determine the factors or model parameters that have the major impact on the performance measures.
- *Production runs and analysis* – Executing the experimental runs to determine the performance measures for each experimental configuration. This step includes the subsequent analysis of the results and the decision of running more experiments if needed.
- *Documentation and reporting* – Documenting the program, the results and conclusions of the simulation study. The documentation should include the conceptual model and the verification and validation to promote model credibility. Frequent progress reports are highly recommended to enhance the successful completion of the simulation study by surfacing misunderstandings early, when the problem can be solved easily. The results of the experiments and the analyses should be reported clearly and concisely in a final report. Including the animation in the final presentation helps also on the model credibility.
- *Implementation* – Putting the results to use. The success of the implementation phase depends on how well the previous steps were implemented. The involvement of the end user through out the simulation process or study, and gaining his/her confidence on the model are critical factors for the success of the project.

2.3. Output Data Analysis

This section describes how to perform the appropriate statistical analysis to the output variables or performance measures of a simulation study. The following references were used for this section: Banks et al. (2001), Charnes (1991), Kelton (1997), Law et al. (1991), Pegden et al. (1995), Sanchez (2001), Sargent (1998), Sargent et al. (2002) and Wilson et al. (2002).

As per Banks et al. (2001), output analysis is the examination of data generated by a simulation and its purpose is to predict the performance of a system or to compare the performance of two or more alternative system designs. This definition suggests the importance of output analysis in any simulation project but despite of this importance, many simulation studies lack of an appropriate statistical analysis for the output variables obtained from the simulation. Law et al. (1991) states that as a matter of fact, a very common mode of operation is to make a single simulation run of somewhat arbitrary length and then to treat the resulting simulation estimates as the "true" model characteristics. This mistake increases the probability of making erroneous inferences with respect to the behavior of the system under study, resulting in wrong conclusions and recommendations.

The need for statistical analysis is driven by the use of random generators to produce the values of the input parameters. As a result, the output data from the simulation will exhibit random variability and the estimates of the performance measures could differ greatly from those of the real system without the proper statistical analysis. Another reason for inadequate statistical results is that the output variables from almost any computer simulation are autocorrelated, meaning a lack of statistical independence which is assumed by classical statistical techniques. As per Sanchez (2001) you cannot treat successive output values as independent observations; if you do so, particularly for short output streams, you are likely to vastly underestimate the system variance and, perhaps provide a biased estimate of the system mean. This can lead to unpleasant surprises when the system is implemented.

The initial conditions of the system at the moment of starting to run the simulation may influence the output data and should be taken into consideration by the analyst. The output data may not be identically distributed if the specified initial conditions are not chosen correctly. This is another reason for inadequate statistical analysis since simulation output is usually conveyed by constructing the confidence interval (or confidence regions in the case of multivariate output) for the estimate of the variable of interest, which is assuming that the data is not only independent but also normally or identically distributed. Traditionally, analysts have expressed the output data from a simulation in the form of confidence intervals in order to have information of both, the center and the spread of the output variable distribution.

The analysis of simulation output data is divided into two categories depending upon the system under study: terminating simulation or non-terminating/steady-state simulation. As per Banks et al. (2001), a terminating simulation is one that runs for some duration of time T_E, where E is a specified event (or set of events) which stops the simulation. Such systems have well-specified initial conditions. The output data for different runs form a terminating simulation model are independent and identically distributed random variables since each run uses independent random numbers and the same initialization rule. Law et al. (1991) specifies that since the initial conditions for a terminating simulation generally affect the desired measures of performance, these conditions should be representative of those for the actual system.

A non-terminating simulation is one for which there is no natural event E to specify the length of a run. The previous definition is given by Law et al. (1991). Banks et al. (2001) defines such a system as one that runs continuously or at least over a long period of time. The simulation of a non-terminating system starts with some initial conditions defined by the analyst and runs for some stopping period T_E specified also by the analyst. In this type of simulation, the analyst is usually more concerned about the steady-state condition of the system. Steady-state can be defined as the long run properties of the system, which are not influenced by the initial conditions of the simulation model at each run. The estimate value of an output variable at the steady-state

condition does not depend on the initial conditions of the simulation; however, the rate of convergence to the steady-state condition does.

The system that is being modeled under this simulation project is a non-terminal system, and as specified above, the interest is to analyze the model under the steady state condition. This is the main reason for focusing from this point on the literature review related to steady-state simulation.

2.3.1. Statistical Analysis for Steady-State Simulation

As per Sargent (2002), systems modeled for steady-state analysis introduce the complexities of: (1) removal of the bias of the imposed initial model state and (2) definition of a sample that admits an accepted estimate of sample variance, which is needed to determine the precision of estimates of steady-state parameters. Removal of the initial state bias or initialization bias[9] is also called the initial transient problem. Estimation of sample variance is usually calculated by applying the method of replications or the batch means method, which are the most common used methods by simulation analysts.

The method used for estimating the sample variance in this study was the replications method. A detailed explanation of this method will be discussed in a further section of this literature review. The following is a brief explanation of some of the methods that have been developed for the estimation of the sample variance including the replications and the batch means methods. Please refer to Kelton (1997), Sanchez (2001) and Wilson et al. (2002).

- *Replication* – The main idea under this method is to make runs as long as possible and then to replicate them. This method will yield independent replications if different sets of random numbers are used for each replication, allowing the use of classical statistics. The problem with this method is that the initial conditions of the model, which are likely to be atypical of the

[9] Initialization bias – refers to the fact that if most (or all) of the output stream is generated during the warm-up period or initial transient, the averages or other summary measures of the data may dramatically overestimate or underestimate the steady-state performance.

system of interest, bias the output at least for a while. Two approaches to overcome the initialization bias are: (1) to plot the output variable with respect to time and find the point where the output stabilizes, then to delete the data prior to that point for each replication; (2) to find a set of initial conditions that are more representative of the system under study. Unfortunately for the latter approach, initial conditions maybe unknown until the simulation is run and some output analysis is done.

- *Batch means* – The main idea under this method is to get as close as possible to the steady-state by running one enormously long run. The problem with this method is that the output data will not be independent (not correlated) since all of the data is coming from one run. The idea is then to split the data into batches that are big enough to guarantee the independency of the data between batches. Classical statistics can then be used as the means of these batches are treated as independent unbiased observations of the behavior of the model under steady-state.
- *Time series model* – The correlated, non-stationary simulation output series can be thought of as a time series. Then a time series model is fit to the data, and the fitted model is used for inferences.
- *Standardize time series* – A process version of the central limit theorem is applied to "standardize" the output series, and methods for statistical analysis have been worked out based on this.
- *Regeneration cycles* – Some simulations return now and then to a state from where they "start over" probabilistically. For instance, if a queue empties out at some point in time, it looks just like it did at the beginning of the simulation (assuming it started empty). This creates independent cycles that are manipulated for statistical analysis.
- *Spectral Analysis* – Estimates of the correlation structure of the process are used to form a variance estimate for statistical analysis.

2.3.2. Transient and Steady-State Behavior

As per Kelton (1991), consider the output data for a particular performance measure from a stochastic process simulation $Y_1, Y_2, \ldots$. Let,

$$F_i(y \mid I) = P(Y_i \leq y \mid I) \tag{2.1}$$

for $i = 1, 2, \ldots$, where y is a real number and I represents the initial conditions at the beginning of the simulation run. The conditional probability $P(Y \leq y \mid I)$ is the probability that the event $\{Y_i \leq y\}$ occurs given the initials conditions I. $F_i(y \mid I)$ is the transient distribution of the output process at discrete time i for initial conditions I. The density functions for the transient distributions corresponding to the random variables Y_{i1}, Y_{i2}, $Y_{i3}, \ldots Y_{ik}$, are shown in Figure 2.1 for a particular set of initial conditions I and increasing time indices $i1$, $i2$, $i3, \ldots ik$, where it is assumed that the random variable Y_{ij} has a density function f_{Yij}. The density f_{Yij} specifies how the random variable Y_{ij} can vary from one replication to another.

If $F_i(y \mid I) \rightarrow F(y)$ as $i \rightarrow \infty$ for all y and for any initial condition I, then $F(y)$ is called the steady-state distribution of the output process $Y_1, Y_2, \ldots$. Strictly speaking, $F(y)$ is only obtained as $i \rightarrow \infty$. In practice, however, there will often be a finite time index, say, $k + 1$, such that the distributions from this point on will be approximately the same as each other; steady-state is figuratively said to start at time $k + 1$ as shown in Figure 2.1. Note that the steady-state distribution $F(y)$ does not depend on the initial conditions; however, the rate of convergence of the transient distributions $F_i(y \mid I)$ to $F(y)$ does.

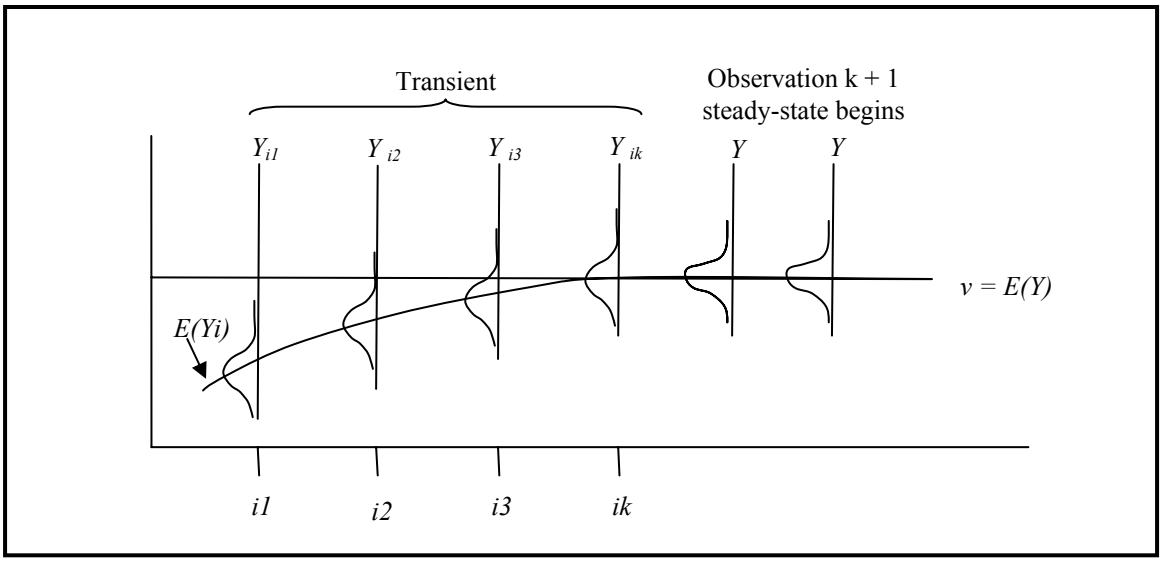

Figure 2.1 Transient and steady-state density functions for a particular stochastic Y_1, Y_2,… and initial conditions *I*. Modified from Law et al. (1991).

2.3.3. Removal of the Initial Bias

Law et al. (1991) states that supposing we want to estimate the steady state mean $v = E(Y)$, which is also generally defined by

$$v = \lim_{i \to \infty} E(Y_i) \tag{2.2}$$

Thus, the transient means converge to the steady-state mean. The problem of the initial transient is that $v \neq E(Y_i)$ during the initial transient period due to the arbitrary initialization conditions specified by the analyst. There are several methods for dealing with this problem: (1) to initialize the simulation to a state more representative of the long run conditions, (2) to delete some number of observations from the beginning of a run and to use only the remaining observations to estimate v (this method is called warming up the model or initial data deletion), and (3) to use (1) and (2) in conjunction.

As per Banks et al. (1991), there are two ways to intelligently specify the initial conditions. The first approach is to collect data of the existing system and to use it to specify initial conditions that are more typical to the real system. Usually, this method requires a large data collection effort and it is impossible to implement if the system

being modeled does not exist. The second approach is to use queuing theory to obtain initial conditions from a second model that has been simplified enough for this purpose.

The warm up or initial data deletion approach is the technique most often suggested for dealing with the problem of the initial transient, as per Law et al. (1991). The idea, as mentioned before, is to delete some number of observations from the beginning of a run and to use only the remaining observations to estimate *v*. For example, given the observations $Y_1, Y_2,\ldots,Y_m$, it is often suggested to use

$$\overline{Y}(m,l) = \frac{\sum_{i=l+1}^{m} Y_i}{m-l} \tag{2.3}$$

($1 \le l \le m$-1) rather than $\overline{Y}(m)$ as an estimator of *v*, which is estimated as

$$\overline{Y}(m) = \frac{\sum_{i=1}^{m} Y_i}{m} \tag{2.4}$$

In general, $\overline{Y}(m,l)$ is expected to be less biased than $\overline{Y}(m)$ because the observations near the beginning of the simulation run has been deleted. The major concern with this method is how to choose the warm up period or deletion amount *l*. As per Law et al. (1991), the following is a graphical procedure due to Welch (1981, 1983), which is the most generally used technique. This procedure is based on making *n* independent replications of the simulation and employing the following four steps:

1. Make *n* replications of the simulation ($n \ge 5$), each of length *m* (where *m* is large). Let Y_{ij} be the *i*th observation from the *j*th replication ($j = 1, 2,\ldots, n$; $i = 1, 2,..., m$).
2. Let $\overline{Y}_i = \sum_{j=1}^{n} Y_{ij} / n$ for $i = 1, 2,\ldots, m$. The averaged process $\overline{Y}_1, \overline{Y}_2,$ has mean $E(\overline{Y}_i) = E(Y_i)$ and variance $Var(\overline{Y}_i) = Var(Y_i)/n$. Thus, the averaged process has the same transient mean curve as the original process, but its plot has only $(1/n)$th the variance.

3. To smooth out the high-frequency oscillations in $\overline{Y}_1$, $\overline{Y}_2$,… (but leave the low frequency oscillations or long run trend of interest), we further define the moving average $\overline{Y}_i(w)$, (where *w* is the window and is a positive integer such that $w \le \lfloor m/2 \rfloor$ as follows:

$$\overline{Y}_i(w) = \begin{cases} \dfrac{\sum_{s=-w}^{w} \overline{Y}_{i+s}}{2w+1} & \text{if i = w + 1,..., m - w} \\ \dfrac{\sum_{s=-(i-1)}^{i-1} \overline{Y}_{i+s}}{2i-1} & \text{if i = 1,..., w} \end{cases} \tag{2.5}$$

Thus, if *i* is not too close to the beginning of the replications, then $\overline{Y}_i(w)$ is just the simple average of $2w+1$ observations of the averaged process centered at observation *i*. It is called a moving average since *i* move through time.

4. Plot $\overline{Y}_i(w)$ for *i* = 1, 2,…, *m* – *w* and choose *l* to be that value of *i* beyond which $\overline{Y}_1(w)$, $\overline{Y}_2(w)$,…appears to have converged.

The major concern with this approach is that the number of replications could be relatively high if the process being modeled is highly variable.

2.3.4. Replication/Deletion Approach

This section presents the replication/deletion approach for obtaining a point estimate and a confidence interval for the steady state mean $v = E(Y)$ of the process Y_1, Y_2,…. There are several conveniences for using this approach:

- Gives reasonably good statistical performance.
- Is the easiest approach to understand and implement.
- Applies to all types of output parameters.
- Can easily be used to estimate several different output parameters.
- Can be used to compare different systems configurations.

As per Law et al. (1991), suppose that we make n' replications of the simulation each of length m' observations, where m' is much larger than the warm up period l determined by the graphical approach discussed in the previous section of this document. Let Y_{ji} be the ith observation from the jth replication (j =1, 2,…, n'; i =1, 2,..., m') and let X_j be given by

$$X_j = \frac{\sum_{i=l+1}^{m'} Y_{ij}}{(m'-l)} \qquad \text{for } j = 1, 2, ..., n' \tag{2.6}$$

Note that only the data from the steady state is being used to calculate X_j. Then the X_j's are independent and identically distributed random variables with $E(X_j) \approx v$, $\overline{X}(n')$ is an approximately unbiased point estimator for v, and an approximate $100(1-\alpha)$ percent confidence interval for v is given by

$$\overline{X}(n') \pm t_{n'-1,1-\alpha/2} \sqrt{\frac{S^2(n')}{n'}} \tag{2.7}$$

where,

$$\overline{X}(n') = \frac{\sum_{i=1}^{n'} X_i}{n'} \tag{2.8}$$

and

$$S^2(n') = \frac{\sum_{i=1}^{n'} \left[X_i - \overline{X}(n')\right]^2}{n'-1} \tag{2.9}$$

Banks et al. (2001) suggests that as a rule of thumb, the length of each replication beyond the deletion point should be at least ten times the amount of the data deleted.

In a steady state simulation, the number of replications can increase in order to achieve a desired confidence interval. Let's say that *h* is the half-width for the confidence interval of $\overline{X}(n')$, and then it is computed as follows:

$$h = t_{n'-1,1-\alpha/2}\sqrt{\frac{S^2(n')}{n'}} \tag{2.10}$$

Assuming that *h* has been determined to be too large after it has been calculated from n' initial replications, then the total number of replications, n^*, required to reduce the half width, *h*, to a desired value, h^*, can be estimated as follows:

$$n^* = [n(h/h^*)^2] \tag{2.11}$$

where [] denotes the value rounded up to the next integer. Please refer to Pegden (1995).

2.3.5. Multivariate Simulation Output Analysis

The following is a brief summary of the literature related to multivariate analysis of simulation output taken from Charnes (1991) and Law et al. (1991). No in-depth discussion will be presented in this section since multivariate techniques for simulation output analysis are not part of the scope of this project. The major reason is that decision making in this project will not be based on confidence interval nor confidence regions (in the case of multivariate output) but in constructing metamodels to specify the problem in the form of mathematical modeling for optimization purposes. The expected result is to define the number of machines and personnel required to achieve a required throughput or output volume, based on the input values of the simulation parameters: yield and testing process time. Ultimately, another expected result is to provide the probability distribution of the model throughput rather than the mean confidence interval.

For most simulation models, several performance measures are of interest to the analyst on a simultaneous fashion. The outputs from these performance measures are usually cross-correlated as well as being auto-correlated. A multivariate statistical technique should be used to analyze the data if cross-correlation is important to the

analyst, thus extracting more information from the simulation model from which to make inferences.

For the case of terminating simulations, one multivariate technique that can be applied is the construction of a joint confidence region, rather than individual intervals for each performance measure of interest. The validity of the procedure rests upon the assumption of multivariate normality and independence of the vector observations taken from each replication.

Usually, the analyst will want to construct individual confidence intervals on the mean of each performance measure of interest. As per Charnes (1991), the analyst must be careful as the overall level of confidence that all intervals with the same nominal confidence level will cover their respective means is less than the nominal confidence level of each interval. Law et al. (1991) warns about the lack of confidence on the conclusions drawn from such a study. The Bonferroni Inequality gives a simple means of setting the individual confidence levels in order to obtain a lower bound on the overall level of confidence.

For the case of steady-state simulations, the analyst is interested in estimating the steady-state parameters of the model, which is the state of statistical equilibrium where the means, cross covariances and auto-covariances are invariant of time. This means that the output vectors are not independent nor identically distributed, and the dependence among the elements and across time is characterized by the auto-covariance function. Please refer to Charnes (1991).

Another method that can be used in the case of steady-state simulations is the multivariate batch-means (MBM). This method attempts to circumvent the autocorrelation problem without losing the information on cross-correlation by grouping the data into approximately uncorrelated batches.

2.4. Comparison of Alternative System Designs

This sections discusses a few of the many statistical methods developed for comparison purposes of two or more system designs on the basis of some performance measures. Comparison of competing alternatives is one of the most important uses of simulation since it gives the analyst the opportunity to compare such alternatives before implementation. The following are some of the references used related to this subject: Banks et al. (2001), Chen (2001), Chick (2000), Goldsman et al. (1998), Jacobson et al. (1999), Law et al. (1991), Nelson et al. (2003a, 2003b), Pegden et al. (1995), Sánchez (2001), and Sargent (1998).

As per Sánchez (2001), the purpose of preparing a simulation model is not to assess the capability of a single system, but to compare one or more systems to a standard level of performance, to compare several systems to one another, or to determine how the performance of one system changes according to particular variants of operating conditions.

Hypothesis tests, confidence intervals, or multiple comparison procedures can be used for comparison of systems against a pre-determined standard. Hypothesis tests can be used in the comparison of means, variances, distributions, and time series of the output variables of a model for each set of experimental conditions. When comparing several systems to one another, selection and ranking procedures can be used. The selection and ranking approach is also useful for comparing systems characterized by different protocols like LIFO and FIFO, or different layout alternatives for a manufacturing facility or process. Surface response methodology can be used if the comparison is based on changes in quantitative input variables. Response surface metamodels seek to approximate the relationship between the input and output variables of the simulation on an analytical fashion, by means of mathematical models developed through regression analysis.

For the purpose of this project, hypothesis tests based on confidence intervals for the mean of the output variables of interest, and response surface based on metamodels will be explained into more details throughout this chapter. Hypothesis tests were used as

part of a sensitivity analysis performed to determine the critical simulation parameters or input variables with respect to the objectives of the project. Metamodels were developed for the experimental region with the purpose of optimizing a particular configuration of interest for the end user. Response surfaces were developed not for optimization purposes as it will be explained later in this chapter, but to confirm that an optimum for a particular configuration was within the experimental region.

Hypothesis tests will be explained in section 2.4.1, construction of confidence intervals for the output mean was discussed in section 2.3.4, metamodels and response surface will be explained in sections 2.5.5 and 2.6.1, respectively. For literature related to ranking and selection procedures, please refer to Chen (2001), Goldsman et al. (1998), Jacobson et al. (1999), Law et al. (1991), Nelson et al. (2003a, 2003b).

2.4.1. Hypothesis Test for Two System Comparison

As previously mentioned, in many occasions the analyst is interested in comparing two systems. The general procedure used for this comparison assumes the number of observations is the same for both systems, which is achieved by performing n replications to each system.

Let's say the names of the two systems under study are *System A* and *System B*. The first step is to state the hypothesis to be tested:

- H_o: Both systems have the same performance based on the output variable being evaluated.
- H_a: Both systems performed different based on the output variable being evaluated.

The second step is to construct the paired-t confidence interval. Please refer to Pegden (1995) for the following explanation of the test; Banks et al. (2001) and Law et al. (1991) can also be referenced.

Let X_{ai} and X_{bi} denote as the observation recorded on the i^{th} replication of *Systems A* and *B*, respectively, and let d_i denote the difference between *Systems A* and *B* on the i^{th} replication (i.e., $d_i = x_{ai} - x_{bi}$). Please refer to Table 2.1

Table 2.1 Theoretical values for t-paired test. Modified from Pegden et al. (1995)

Replication	System A	System B	Difference
1	X_{a1}	X_{b1}	d_1
2	X_{a2}	X_{b2}	d_2
⋮	⋮	⋮	⋮
n	X_{an}	X_{bn}	d_n

The comparison of the two systems is approached by developing a confidence interval on the expected value of d_i, which for purpose of this explanation will be named δ. The resulting confidence interval is referred to as a paired-t confidence interval.

The statistical convenience of using this method is the following: (1) the assumption that the variances of each system are equal (as in the two sample-t method) is not necessary, (2) there is no need for assuming that X_{ai} and X_{bi} are independent. We only need to assume that within *System A* the observations are independent as well as in *System B*, which is fair considering that each observation within a system is coming from different replications.

In order to compute the confidence interval, we begin by estimating $\bar{d}$, $s(d)$, and $s(\bar{d})$, as follows:

$$\bar{d} = \sum_{i=1}^{n} \frac{d_i}{n} \tag{2.12}$$

$$s(d) = \sqrt{\sum_{i=1}^{n} \frac{(d_i - \bar{d})^2}{n-1}} \tag{2.13}$$

$$s(\bar{d}) = \frac{s(d)}{\sqrt{n}} \tag{2.14}$$

The half-width for $1-\alpha$ confidence interval on δ centered at $\bar{d}$ is then given by

$$h = t_{n-1,1-\alpha/2} s(\bar{d}) \tag{2.15}$$

If the two systems perform identically, then the expected value for $\bar{d}$ is 0. Therefore, if the computed confidence interval contains 0, there is not enough evidence to reject H_0 and the analyst cannot reliably state that both systems are different. However, if the interval does not contain 0, then H_0 can be rejected and the analyst can assert with appropriate confidence level that both systems perform different with respect to output variable or performance measure X.

A second approach for forming the confidence interval of δ when the number of observations from both systems is not equal is the two sample-t confidence interval. As in the previous approach, there is no need for assuming that X_{ai} and X_{bi} are independent, but the variances of each system must be equal. However, n_1 and n_2 can be different. For more detail on this approach, please refer to Law et al. (1991) and Devore (1982).

2.5. Experimental Design

As per Montgomery (2001), experimental design is a test or series of tests in which purposeful changes are made to the input variables of a process or system so that we may observe and identify the reasons for changes that may be observed in the output response. In the context of simulation, the system or process is a computer model of the real system under study, either actual or planned. This section is an introduction to the use of design of experiments when the experiment is the execution of a computer simulation.

In experimental design terminology, the input parameters and structural assumptions of the simulation model are called factors, and the output performance measures are called responses. Usually, there are several responses or performance measures of interest. Factors may be either quantitative or qualitative, and in some

occasions the distinction is not clear. Factors can also be classified as being controllable or uncontrollable, depending on whether they represents action options to managers of the corresponding system of interest. Please refer to Law et al. (1991).

Usually, at the early stages of experimentation, the analyst is concerned about which factors are important and how they might affect the responses. As the analyst understanding of the model's behavior progresses, the interest on experimentation turns from identifying critical factors to defining the optimal combinations of factor levels that maximize or minimize a response of interest. At this stage, the use of metamodels and response surface methodologies can be of great value. Once metamodels are constructed, the estimation of the gradient could also be of interest to the analyst for purposes of quantifying how the responses react to small changes in the quantitative factors as well as for optimization purposes.

This section explains into more details the use of sensitivity analysis, 2^k full factorial designs, 2^{k-p} fractional factorial designs, central composite designs and metamodels.

The following are the references used for this section: Banks et al. (2001), Barton (1998), Chik (2000), dos Santos et al. (1999), Ivanova et al. (1999), Kelton (1997), Kelton et al. (2003), Kleijnen et al. (2001), Law et al. (1991), Montgomery et al. (2001), Sánchez (2001), and Wan (2003).

2.5.1. Sensitivity Analysis

As per Kleijnen et al. (2001), sensitivity analysis is the systematic investigation of the reaction of the simulation responses to extreme values of the model's input or to drastic changes in the model's structure. Such an analysis helps identify the most important factors in a simulation study. Kelton (1997) states that a direct way to address this is to make two sets of runs, one at the original value and another one at the changed value of the input parameter, and then to look at the difference. Law et al. (1991) specifies that if the output is sensitive to some aspect of the model, then that aspect must be modeled carefully.

It is important to control the randomness in the simulation when using sensitivity analysis. Otherwise, the effect of changing one input parameter may be confounded with other changes in the model as a result of not using the same set of random numbers for the elements of the model that were not changed.

Sensitivity analysis can be used during different steps of the simulation study. It is highly recommended during the validation step; it can also be used during the early stages of the experimental step to determine critical factors, and during the optimization phase.

Usually, only one factor is changed at a time during sensitivity analysis, but if the analyst is interested in changing two or more factors at the same time, then statistical design of experiments is the most appropriate tool to use. Sensitivity analysis was used in this project for the purpose of identifying input parameters of the simulation that should be considered as experimental factors as well as for defining the experimental levels for those factors.

2.5.2. 2^k Factorial Designs

One of the main objectives of experimental design in simulation is to determine how changes in the factors or input parameters of the model affect the responses or performance measures of the model. In the case in which the model has only one factor, the idea is to run the simulation at different levels or values of the factor. Law et al. (1991) suggests that it may be useful to form a confidence interval of the expected response at each of the factors levels, and to graph the response as a function of the factor levels.

Suppose that a system with k ($k \geq 2$) factors or input parameters of interest is being modeled. The analyst is interested in understanding, not only how each factor may affect the responses, but also how each factor interact within each other. One way to address this problem is through experimental design, in particular 2^k factorial design. This experimental design requires from the analyst to choose two levels for each factor

and then calls for simulation runs at each of the 2^k possible factor-level combinations, which are known as design points or treatments.

The statistical model for a 2^k design would include k main effects, c_2 two-factor interactions, c_3 three-factors interactions,…, and one k-factor interaction. Where,

$$c_2 = \binom{k}{2} \tag{2.16}$$

and

$$c_3 = \binom{k}{3} \tag{2.17}$$

The complete model would contain $2^k - 1$ effect. As per Montgomery (2001), the treatment combination may be written in standard order by introducing the factors one at a time, with each new factor being successfully combined with those that precede it. For example, the standard order for a 2^4 design is (1), *a, b*, *ab*, *c, ac*, *bc*, *abc*, *d*, *ad*, *bd*, *abd*, *cd*, *acd*, *bcd*, *abcd*. Refer to Table 2.2.

Table 2.2 Design matrix for a 2^k factorial design. Modified from Law et al. (1991) and Montgomery (2001).

Design Point	Treatment Combination	Coded Factors					Response Variable (*Yij*)				
		A	*B*	*C*	…	*K*	Rep. 1	Rep. 2	…	Rep. n	Total
1	*(1)*	-	-	-	…	-	*Y11*	*Y12*	…	*Y1n*	*Y1*
2	*a*	+	-	-	…	-	*Y21*	*Y22*	…	*Y2n*	*Y2*
3	*b*	-	+	-	…	-	*Y31*	*Y32*	…	*Y3n*	*Y3*
4	*ab*	+	+	-	…	-	*Y41*	*Y42*	…	*Y4n*	*Y4*
5	*c*	-	-	+	…	-	*Y51*	*Y52*	…	*Y5n*	*Y5*
6	*ac*	+	-	+	…	-	*Y61*	*Y62*	…	*Y6n*	*Y6*
7	*bc*	-	+	+	…	-	*Y71*	*Y72*	…	*Y7n*	*Y7*
8	*abc*	+	+	+	…	-	*Y81*	*Y82*	…	*Y8n*	*Y8*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
k	*abc…k*	+	+	+	…	+	*Yk1*	*Yk2*	…	*Ykn*	*Yk*

Montgomery et al. (2001) describes the following procedure for the statistical analysis of the 2^k design. It is important to mention that even though the procedure is not described in the context of computer simulation experiments, the procedure is for generic

application and for any number of *k* factors. This was the procedure followed during the evaluation and analysis stage for this project.

1. Estimate the factor effects and examine their signs and magnitudes. This gives the experimenter preliminary information regarding which factors and interactions may be important, and in which directions these factors improve the response.
2. Form the initial regression model or metamodel in the simulation slang. The full model, that is, all main effects and interactions, provided that at least one of the design points has been replicated, is the following:

$$Y = \beta_0 + \beta_A A + \beta_B B + ... + \beta_K K + \beta_{AB} AB + ... + \beta_{AB...K} AB...K + \in \qquad (2.18)$$

   where coded factors are represented by A, B,..., K, $\beta's$ are the regression coefficients, and $\in$ is a random error with mean zero and constant variance σ^2.
3. Perform an analysis of variance to formally test for significance of main effects and interactions. Refer to page 243 in Montgomery (2001) for general form of an analysis of variance for a 2^k factorial design with *n* replicates.
4. Remove any non-significant variables from the full model. There is no need to create a new model with only the significant factors since by design these experiments have no presence of multicollinearity[10]. These experimental designs are orthogonal or nearly orthogonal; as per Myers et al. (1995) multicollinearity is not a problem in this type of designs. This also applies to models derived from 2^{k-p} fractional factorial and central composite experimental designs.
5. Residual analysis to check for model adequacy and to check the following assumptions related to the analysis of variance: mean value of the residuals is cero, residuals are normally distributed, residuals are independent, and

[10] Multicollinearity – when there is near-linear dependency among the regressor variables of the regression model; it can have serious effects on the estimates of the model parameters and on the general applicability of the final model.

variance of the residuals is constant. Montgomery (2001) suggests the following graphical analysis of residuals:

- *Normal probability plot of residuals*
- *Plot of residuals in time sequence*
- *Plot of residuals versus fitted values*

6. Graphical analysis that usually consists of either main effects or interaction plots, or response surface and contour plots.

In order to estimate the effect of the main factors and interactions (*e)*, and the sum of squares for an effect (*SSE*), it is necessary to determine first the contrast associated with that effect. In general, we determine the contrast for effect $e_{AB...K}$ by expanding the right hand side of

$$\text{Contrast}_{\text{AB...K}} = (\text{a} \pm 1)(\text{b} \pm 1)...(\text{k} \pm 1) \tag{2.19}$$

In expanding Equation 2.19, ordinary algebra is used with "1" being replaced by (1) in the final expression. The sign in each set of parentheses is negative if the factor is included in the effect and positive if the factor is not included.

To illustrate the use of equation 2.19, consider a 2^2 factorial design. The contrast for *AB* would be

$$Contrast_{AB} = (a-1)(b-1) = (1) - a - b + ab \tag{2.20}$$

Note that the same results can be obtained using Table 2.2. Simply multiply the "Treatment Combination" column by the column of the factors of the effect of interest. This is a relatively easy way of calculating the effects for small values of *k*; but for large number of factors is better to calculate the contrast. For the numeric value of the contrast, just replace the codification in standard order for the treatment combination by the corresponding response value under the column "Total" of Table 2.2. Assuming that Y_i is the summation of the values observed for all replications at the design point *i*, then

$$Contrast_{AB} = Y_1 - Y_2 - Y_3 + Y_4 \tag{2.21}$$

The effects and sum of squares of the effects can be estimated once the contrast has been calculated, by the use of the following equations:

$$e_{AB\ldots K} = \frac{2}{n2^k}\left(Contrast_{AB\ldots K}\right) \tag{2.22}$$

and

$$SSE_{AB\ldots K} = \frac{1}{n2^k}\left(Contrast_{AB\ldots K}\right)^2 \tag{2.23}$$

respectively, where *n* denotes the number of replicates.

A potential concern with the use of 2^k factorial designs is the assumption of linearity. Even though the full regression model or metamodel described on Equation 2.18 has the capability of presenting some curvature in the response function as result of the interaction terms added to the main effects, there could be situations in which the curvature in the response function is not adequately modeled. In such cases, a second order or quadratic response surface model will be required. Adding center points to a 2^k factorial design is helpful in identifying curvature. The presence of curvature is identified calculating the single-degree-of-freedom sum of squares for pure quadratic curvature given by

$$SS_{Pure\,\text{quadratic}} = \frac{n_F n_C \left(\bar{y}_F - \bar{y}_C\right)^2}{n_F + n_C} \tag{2.24}$$

where n_F is the number of factorial design points, n_C is the number of observations at the center point, $\bar{y}_F$ is the average of the runs at the factorial point, and $\bar{y}_C$ is the average of observations at the central point. Refer to Montgomery (2001) for more details on factorial designs with center point.

Kelton et al. (2003) states that another difficulty with 2^k factorial designs is that if the number of factors becomes even moderately large, the number of runs explodes exponentially. A way around this is to use what are known as fractional-factorial designs in which only a fraction of all the possible factor-combinations are run.

2.5.3. 2^{k-p} Fractional Factorial Design

One of the concerns with 2^k factorial designs is that the number of design points grows exponentially as the number of factors increases, thus outgrowing the resources available by the analyst and increasing the cost of the simulation study. As per Montgomery (2001), if the experimenter can reasonably assume that certain high-order interactions are negligible, information of the main effects and low-order interactions may be obtained by running only a fraction of the full factorial experiment.

As per Law et al. (1991), the 2^{k-p} fractional factorial design is constructed by choosing a certain subset of size 2^{k-p} of all possible design points, and then running the simulation for only those chosen points. Fractional factorial designs are used as screening experiments. These are experiments in which many factors are considered and the objective is to identify those factors that have large effects. Screening experiments are usually performed at the early stages of the project and the factors identified as critical to the responses of interest are then investigated more thoroughly in subsequent experiments. The fractional factorial designs are based on three main ideas:

- *The sparsity of effects principle* – when there are several variables, the system or process is likely to be driven primarily by some of the main effects and low-order interactions.
- *The projection property* – Fractional factorial designs can be projected into stronger (larger) designs in the subset of significant factors.
- *Sequential experimentation* – It is possible to combine the run of two or more fractional factorials to assemble sequentially a larger design to estimate the factor effects and interactions of interest.

The following explanation of the 2^{k-p} fractional factorial design has been taken from Montgomery (2001). These designs require the selection of p independent generators. The defining relation for the design consists of the p generators initially chosen and their 2^p-p-1 generalized interactions. Care should be exercised in choosing the

generators so that effects of potential interest are not liased with each other. Each effect has 2^p-1 aliases. For moderately large values of k, it is common to assume that higher-order interactions, such as interactions within 3 factors or higher are negligible. In many situations, this is a practical assumption that greatly simplifies the alias structure.

It is important to select the p generators for a 2^{k-p} fractional factorial design in such a way that the best possible alias relationship is obtained. A reasonable criterion is to select the generators such that the resulting 2^{k-p} design has the highest possible resolution. Law et al. (1991) states that in simulation, resolution IV designs may be inadequate and resolution III designs should probably be avoided except in desperation.

As per Montgomery (2001), sometimes, resolution alone is insufficient to distinguish between experiments. Consider the case of a study in which the possible fractional factorial designs that can be used have the same resolution. In such a case, the design with the least number of lower-order interactions should be taken.

The effects of a 2^{k-p} fractional factorial design is estimated by

$$e_i = \frac{2(Contrast_i)}{N} = \frac{Contrast_i}{\left(\frac{N}{2}\right)} \tag{2.25}$$

Where the $Contrast_i$ is found using the plus and minus signs of the 2^{k-p} design matrix, and $N = 2^{k-p}$ is the total number of observations. The 2^{k-p} design allows only $2^{k-p} - 1$ effects, and their aliases, to be estimated.

For more information related to 2^{k-p} fractional factorial design and its applications to simulation, refer to Law et al. (1991). Nevertheless, for a more detailed explanation of this type of experiment but with focus on experimental design literature rather than simulation, Montgomery et al. (2001) is a good starting point.

With respect to simulation theory, Law et al. (1991) states that for a moderate number of factors, 2^{k-p} fractional factorial designs are able to indicate which factors are irrelevant and can be fixed at some reasonably value, and omitted from further

consideration. But in the situation where too many factors are present for even a highly fractionated factorial design to be workable, other screening designs can be used. Some of these experiment designs are: Plackett-Burman designs, supersaturated designs, random-balance designs (which is one kind of supersaturated design), systematic supersaturated designs and group-screening designs. For more references on this subject, please refer to Law et al. (1991). For a practical application of a group-screening design, refer to Ivanova et al. (1999). Finally, refer to Wan et al. (2003) for a theoretical explanation of a new type of factor-screening design.

2.5.4. Central Composite Design

As previously mentioned in section 2.4.2, a potential concern with the use of 2^k factorial designs is the assumption of linearity. In the case in which the curvature in the response function is not adequately represented by a model of first order (linear model), such as the one described in Equation 2.18, a second order or quadratic response surface model, such as the one described in Equation 2.26 will be required.

$$Y = \beta_0 + \beta_A A + ... + \beta_K K + ... + \beta_{A...K} A...K + \beta_A A^2 + ... + \beta_K K^2 + \in \qquad (2.26)$$

As per Montgomery (2001), an experiment design that will fit this model is a 2^k factorial design (or fractional factorial of resolution V) with n_F runs, $2k$ axial or star runs, and n_C center runs. Such a design is called central composite design.

Usually, central composites are designed after a 2^k factorial design failed to fit a first order model by adding the axial runs to allow the quadratic terms to be incorporated into the model. Central composites are very efficient and are the most popular class of designs for fitting the second order models. There are two parameters in the design that must be specified: the distance α of the axial runs from the design center, and the number of center points n_C. The choice of α is dictated primarily by the region of interest. Refer to Montgomery (2001) for possible values of α and more details about this type of experiment design.

2.5.5. Metamodels

A simulation model can be thought of as a process in which input parameters are transformed into output variables or responses. In that sense, the relationship between the parameters and the response of the simulation can be expressed as an algebraic or mathematical function, which in turn is a model of the simulation model itself. This algebraic function is developed with the use of regression analysis from data taken from experimental designs, in which the dependent variable of the regression is the output variable or response of the simulation, and the independent variables are the factors or input parameters. Kelton et al. (2003), mentions that since the regression model is an approximation of the simulation model, the regression is a "model of a model" and so is called a metamodel.

Kelton et al. (2003) states that the metamodel can serve several different purposes:

- Partial derivatives can be taken of the metamodel to estimate the effect of small changes in the factors on the output response, and any interaction that might be present as modeled, would show up naturally.
- The analyst can use the metamodel as a proxy for the simulation, and very quickly explore many different input-factor-level combinations without having to run the simulation.
- The metamodel can be optimized (maximize or minimize) to give a sense of where the best input-factor-combinations might be.

Barton (1998) specifies that the major issues in metamodeling include: (1) the choice of a functional form for the regression model, (2) the experimental design and (3) the assessment of the adequacy of the metamodel.

For the purpose of this project, it was decided to use regression models of first (linear) and second order (quadratic), which are the base for 2^k factorial and 2^{k-p} fractional factorial designs, and central composite designs, respectively. Refer to Equation 2.18 and Equation 2.26; $\beta_{j's}$ are the coefficients of the regression models and could easily be

estimated as follows (for the 2^k factorial, 2^{k-p} fractional factorial and central composite designs):

$$\beta_0 = \overline{Y} \tag{2.27}$$

$$\beta_i = \frac{\mathrm{e_i}}{2} \tag{2.28}$$

where β_0 is the grand average of all observations, and β_i are one-half of the corresponding factor effect estimate (refer to equations 2.22 and 2.25).

As for the metamodel adequacy, the test for lack of fit can be done. The main objective for the lack of fit test is to determine whether a specific type of regression function adequately fits the data. The lack of fit test is done as follows:

$$F_0 = \frac{SS_{LOF} \big/ (m-p)}{SS_{PE} \big/ (n-m)} \tag{2.29}$$

where SS_{LOF} and SS_{PE}, are the sum of squares due to lack of fit and the sum of squares due to pure error, respectively; n is the total number of observations, m is the number of levels for each factor, and p is the number of parameters that will be estimated. SS_{LOF} and SS_{PE} are calculated using equations 2.30 and 2.31.

$$SS_{LOF} = \sum_{i=1}^{m} (n_i - 1) = n - m \tag{2.30}$$

$$SS_{PE} = \sum_{i=1}^{m} \sum_{j=1}^{n_i} \left(y_{ij} - \bar{y}_i \right)^2 \tag{2.31}$$

Where n_i is the number of observations at the i^{th} level of factor X. If the true regression is linear, then F_0 follows the $F_{m-p,n-m}$ distribution. Therefore, if $F_0 > F_{\alpha,m-p,n-m}$, it can be concluded that the true regression function is not linear.

For more details on metamodels and its application to simulation, please refer to Banks et al. (2001), Barton (1998), Kelton (1997), dos Santos et al. (1999), Kelton et al.

(2003), and Law et al. (1991). For more detailed information on regression analysis to Montgomery (2001) and Neter et al. (1996).

2.6. Simulation Optimization

Fu (2001) defines simulation optimization as the optimization of performance measures based on outputs from stochastic (primarily discrete events) simulations. As in any optimization problem, the primary components are also present: input and output variables, objective function, and constraints. The input variables or parameters may be either quantitative or qualitative (categorical). The quantitative input variables may assume continuous or discrete values. Usually, there are multiple performance measures or output variables of interest, which are combined into one objective function by the use of appropriate weights, or by including them as constraints. As per April et al. (2003), The main goal of simulation optimization is to find the combination of factor levels that minimizes or maximizes the objective function subject to the constraints imposed on factors and/or responses. The factors of interest are the ones that have the greatest effect on the responses as determined by the experimental designs.

As per Banks (2001), the main approaches for simulation optimization are categorized by algorithms that:

- *Guarantee a pre-specified probability of correct selection* – The main purpose of these methods is to select the best alternative from a set of pre-defined systems, as in ranking and selection methods. In that sense, they are not optimization procedures per se, since they lack the search feature.
- *Guarantee asymptotic convergence* – These procedures are usually for quantitative-continuous input parameters that guarantee convergence to the global or a local optimal solution.
- *Optimal for deterministic counterpart* – The idea is to find the optimal solution by the use of an algorithm typical to deterministic optimization (mathematical programming formulations). This approach is suggested only if the simulation performance of each design or alternative can be evaluated with certainty, in other words, there is no statistical error or sampling

variability. In order to guarantee this, the analyst must make sure that enough simulation effort (replications or run length) is expended.

- *Robust heuristics* – These are heuristic procedures that have been developed for deterministic optimization, that have shown to be effective for difficult practical problems, even though they do not guarantee the optimum. Some of these heuristics use randomness as part of their search strategy. Nevertheless, it is still important to make sure that enough simulation effort is expended.

The challenge associated with finding the optimal combination of input parameters is that the relation (algebraic function) between the parameters and the response is unknown, which means that it has to be estimated. Swisher et al. (2000), states that the stochastic nature of the simulation output complicates the optimization problem. This may require multiple simulation runs (replicates) or long simulation runs to assure no misled of the optimization algorithm by the variability of the response.

The simulation optimization techniques can be classified by the input parameters of the model since they are applied depending upon the type of input parameters, in particular, continuous or discrete, both being quantitative type of variables. In the case of qualitative or categorical variables, ranking and selection can be used as previously mentioned in section 2.4. The following is a brief explanation of several of these techniques. The following have been used as reference: Andradóttir (1998), April et al. (2003), Banks et al. (2001), Fu (2001), Jin et al. (2003), Law et al. (1991), Stuckman et al. (1991), and Swisher et al. (2000).

2.6.1. Continuous Input Parameter Methods

As per Swisher et al. (2000), in the case of continuous input parameters, methods may be classified as either gradient-based or non-gradient-based. In the case of the gradient based, the analyst is looking at the partial derivatives of the estimated response function (regression model) with respect to the input parameters. The gradient is important since it gives the sensitivity of the response variable to small changes in the input parameters, thus providing the direction in which to search for the optimum. For references on non-gradient-based approaches, see Swisher et al. (2000).

Some of the gradient based algorithms are: the finite differences gradient estimation approach, perturbation analysis, the likelihood ratio method, stochastic approximation algorithms, and sequential surface response methodology. As per Andradóttir (1998), a difficulty implementing the finite differences gradient estimation approach is that the estimators obtained usually have a large variance, which requires the use of common random numbers for the different simulation runs or replicates. Whereas perturbation analysis and likelihood ratio method require only a single run, producing estimators with the desire statistical conditions in terms of variability. As per April et al. (2003), stochastic approximation algorithms attempt to mimic the gradient search method used in deterministic optimization by estimating the gradient in the objective function in order to determine a search direction. In the case of surface response methodology, the goal is to obtain an approximate functional relationship (regression model) between the input variables and the output objective functions, as stated by Fu (2001). When this is done on the entire (global) domain of interest, in principle, appropriate deterministic procedures can be applied to obtain an estimate of the optimum. However, in practice, sequential response surface methodology is used rather than deterministic approaches when optimization is the main objective of the study, mainly due to the concern of misleading a deterministic based algorithm by the variability of the stochastic process of the simulation model. In the case of sequential response surface methodology, the metamodel characterizes the objective function in a local area, rather than in the entire solution space; then, this local response surface is used to determine a search strategy by means of moving in the direction of the estimated gradient.

2.6.2. Discrete Input Parameter Methods

Several techniques are available for optimization simulation when the input parameters are discrete. Ranking and selection and multiple comparison procedures can be applied when the set of alternatives is finite and small. On the other hand, if the set of possible alternatives is infinite or large, ordinal optimization, simulated annealing, genetic algorithms, tabu search and random search can be applied.

As previously mentioned, ranking and selection is used when the objective is to select the best alternatives from a set of finite pre-defined alternatives. As per Swisher et al. (2000), the objective is to determine the combination of input parameters that minimizes the objective function. This is done in two phases: (1) looks to determine a subset of input combinations that most likely minimizes the objective functions, and (2) select the best alternative. Multiple comparison procedures attempt also to identify the best system from a set of possible alternatives but do not guarantee a decision.

In the case where there are a large number of feasible solutions, the following can be said:

- *Ordinal optimization* - Focuses on finding a good alternative, rather than the best one.
- *Simulated annealing* – Emulate the annealing process for crystalline solids, where a solid is cooled very softly from a very high temperature. Simulated annealing is a variation of local search for deterministic problems.
- *Genetic algorithms* – Mimics the evolutionary behavior of biological systems to create subsequent generations that guide the search towards or near optimal solutions.
- *Tabu search* – It is a variation of local search that incorporates two strategies: adaptive memory and responsive exploration, to guide the search near optimal solutions by dynamically managing a list of forbidden moves (from where the process derived its name).
- *Random search* – These algorithms moves iteratively between neighboring feasible points in search of the optimal solution.

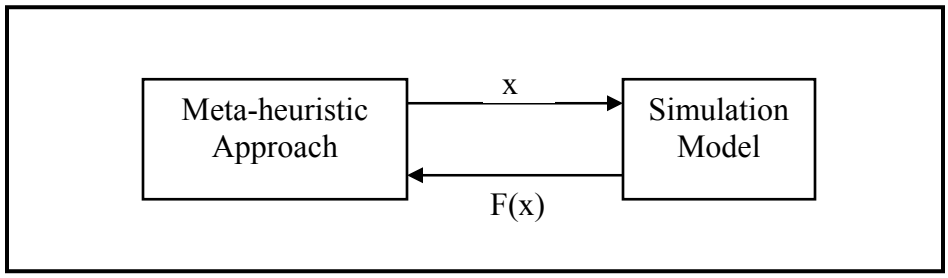

Figure 2.2 Meta-heuristic approach for simulation optimization.

All of the above methods for a large number of alternatives are known as meta-heuristic methods to simulation optimization, in which the simulation is run to define the function that relates the input parameters with the response variable, and then feedback into the meta-heuristic method to define a new set of alternatives and starting the process all over again.

2.7. Solver Methodology

Frontline Systems is the company that developed the Solver in Excel for Microsoft. They offer different platforms compatible with Microsoft's Excel and Visual Basic for Applications, and even C/C++ and other languages. The recommended platform depends on the type and size of the problem, which in the case of this project, turned out to be the standard Microsoft Excel Solver.

The standard Microsoft Excel Solver can solve linear programming up to 200 decision variables with a basic implementation of the primal Simplex method; mix integer programming with a basic implementation of the Branch and Bound method, but limited to 10 to 20 integer variables; smooth non-linear optimization using the Generalized Reduced Gradient (GRG) method up to 200 decision variables and 100 constraints in (addition to bounds on the variables). In the case of this project the optimization models were of the smooth non-linear optimization type, thus developing the optimization tool with Excel Solver was of practical value for the end user. A smooth non-linear programming is one in which the objective function or at least one of the constraints is a continuous non-linear function of the decision variables. Quadratic programming is a special case of the non-linear optimization problem.

The GRG method is an algorithm well known in the mathematical programming arena for solving optimization problems in which non-linear constraints and arbitrary bounds on the decision variables are allowed. The form is: Max f(x): h(x)=0, L <= x <= U, where h has dimension m. The method supposes we can partition x = (v, w) such that:

- v has dimension m (and w has dimension n-m);
- the values of v are strictly within their bounds: L_v < v < U_v

- grad_v[h(x)] is nonsingular at x = (v,w).

For any w there is a unique value, v(w), such that h(v(w),w)=0, which implies that $dv/dw = \text{grad_vh}(x)^{-1}\ \text{grad_w}[h(x)]$. The idea is to choose the direction of the independent variables to be the reduced gradient: grad_w[f(x) - yh(x)], where $y = dv/dw = (\text{grad_vh}(x)^{-1})\text{grad_w}[h(x)]$. Then, the step size is chosen and a correction procedure applied to return to the surface, h(x) =0.

The code used by Microsoft Excel Solver is the GRG2, which was written in ANSI Fortran. It seeks a feasible solution first (if one is not provided) and then retains feasibility as the objective is improved. It uses a robust implementation of the BFGS quasi-Newton algorithm as its default choice for determining a search direction. A limited-memory conjugate gradient method is also available, permitting solutions of problems with hundreds or thousands of variables. The problem Jacobian is stored and manipulated as a dense matrix, so the effective size limit is one to two hundred active constraints (excluding simple bounds on the variables, which are handled implicitly). The GRG2 software may be used as a stand-alone system or called as a subroutine. The user is not required to supply code for first partial derivatives of problem functions; forward or central difference approximations may be used instead. As per Boesel (2001), a special development of recent interest in simulation optimization is the integration of meta-heuristic search with classical non-linear optimization such as the state-of-the-art GRG2.

There are other Microsoft Excel compatible platforms that can solve the same type of optimization problems described above but with more complexity in terms of the number of variables and restrictions. In addition, those platforms can solve the following type of optimization problems: quadratic programming, constraint programming, global optimization, and non-smooth optimization.

In summary, other algorithms used to solve optimization problems are: active set methods, Interior Point or Newton-Barrier method, Cut Generation methods, Strong Branching method, Preprocessing and Probing methods, integer heuristic methods, multi-start methods for global optimization, Sequential Quadratic Programming method,

interval methods, Continuous Branch and Bound method, Genetic and Evolutionary algorithms, Tabu Search and Scatter Search.

2.8. Summary

This chapter covered the most relevant subjects discussed in the computer simulation literature with focus given to the output analysis, experimental designs and optimization.

It was mentioned that output analysis is the examination of data generated by a computer simulation with the purpose of predicting the performance of the system or performing a comparison of the performance of two or more system designs. The need for statistical analysis is driven by the use of random generators to produce the values of the input parameters. As a result, the output data from the simulation will exhibit random variability and the estimates of the performance measures could differ greatly from those of the real system without the proper statistical analysis.

Statistical analysis for the non-terminating/steady state simulation was discussed in detail. In particular, the chapter covered the transient or steady state behavior as well as the replication/deletion approach for calculating the estimates of the performance measures of interest. Hypothesis test was discussed for the comparison of alternative systems designs.

Experimental designs with respect to simulation were also discussed, in particular 2^k factorial, 2^{k-p} fractional factorial and central composite. Focus was given to the calculation of the effects, the sum of square errors, analysis of variance and lack of fit test, and regression analysis for metamodeling purposes.

The chapter covered also the simulation optimization literature with a summary of the most common techniques. It was mentioned that the applicable technique will depend on the type of input parameters. A brief summary was given with respect to Microsoft's Excel Solver and the algorithms used based on the type of optimization problem.

The following chapter (III) will discuss the methodology followed for the execution of this project. In particular, it will cover the use of sensitivity analysis, experimental designs, and regression analysis for metamodeling purposes. It will also discuss the optimization tool developed with the use of Microsoft's Excel and Visual Basic for Applications, and the methodology followed for validation purposes. Chapter IV will discuss the results and statistical analysis, and Chapter V will present the conclusions, and recommendations in terms of future research.

CHAPTER III. Methodology

3.1. Introduction

This section describes the methodology followed for the execution of this project. As previously mentioned on sections 1.4 and 1.5, the objectives were defined as related to the practical application of the problem being addressed, and the academic interest in the output analysis of computer simulation models. Related to the practical application, the objective was to provide the end user with a tool that allows him/her to understand the behavior of the system for the whole spectrum of alternatives or possibilities as defined by the yield and testing process time. In terms of the academic interest, there were two main objectives: (1) the use of design of experiments, metamodels, and optimization, and (2) to demonstrate that the use of such tools as part of simulation modeling and analysis yields better results than the traditional trial and error approach. The methodology described below, pretends to achieve these objectives.

The section has been divided into four subsections: (1) Sensitivity analysis (2) Reducing the problem with experimental designs, (3) Development of metamodels, and (4) Optimization tool. Section (1) describes the methodology followed to determine the critical factors or input parameters with respect to the response variables or performance measures of interest. Section (2) covers the application of experimental designs in order to reduce the number of critical factors to be considered in the metamodels. Section (3) explains the use of experimental designs in order to establish the metamodels, and finally, section (4) describes the optimization tool developed, having in mind the end user.

Whenever appropriate, the section refers to the Appendixes for backup information or data. It is important to mention at this stage, that ARENA[11] was the simulation software used for modeling and output analysis purposes. Please refer to Appendix B to view the simulation model in terms of the SIMAN codes.

[11] ARENA – A simulation software that combines the ease of use of high level simulators with the flexibility of SIMAN, and even all the way down to general purpose languages. SIMAN is a simulation language.

3.2. Sensitivity Analysis

Sensitivity analyses were done with respect to two main objectives: (1) to validate the warm-up period defined during the simulation analysis of the model for the actual (real) system, and (2) due to the high number of input parameters at the beginning of the study, to identify with the minimum simulation time effort the parameters that should be considered as experimental factors and their respective experimental levels. The methodology followed was practically the same for both cases. In general, the objective was to vary one factor at a time to determine its impact on the responses of the simulation. Any input parameter that statistically impacted by means of hypothesis testing at least one performance measure was considered as an experimental factor. It is important to mention that even though the warm-up period can not be defined as an input parameter of the model per se; the methodology followed can still be described as sensitivity analysis.

3.2.1. Validation of the Warm-up Period

As previously mentioned on section 1.3, the simulation model (actual/real system) was verified and validated following the suggested methodology by the simulation literature (refer to Appendix A for a summary). For this reason, verification and validation of the model was not in the scope of this project. Nevertheless, it was considered as a best practice to do some kind of not-in-depth confirmation of the verification and validation. Considering that during the previous verification and validation processes, the simulation responses were compared to the performance measures of the real system, it was decided to do the confirmation by means of the warm-up period only. In other words, if the responses were statistically proven to be the same as with a longer warm-up period, then there were no reasons to doubt about the previous verification and validation of the model.

The validation of the warm-up period was done following the next steps:

1. The simulation model was run with the same conditions (input parameters, simulation run-length, warm-up period, and number of runs) as it was run

before for validation purposes. Please refer to Appendix C to view the level or value at which the input parameters were setup. The simulation run-length was 19,320 minutes, the warm-up period was 920 minutes, and the number of runs was 100. 18,400 minutes per replicate, were used for data collection and output analysis purposes. This is the equivalent to 920 minutes per day (2 shifts of 460 minutes each) on a month of 20 production days. Data from all of the output variables from the model were recorded. This model was named as the "Actual System".

2. All performance measures except the "Throughput" were assumed to be independent and identically distributed without doing any statistical validation. This is a fair assumption based on the replication/deletion method (refer to section 2.3.4.). As for the "Throughput", data was assumed to be independent as they are the result of different replicates, but a normality test was performed using Minitab to validate that they were identically distributed.
3. Confidence intervals for each simulation responses or output variables (performance measures) were built with the Output Analyzer option from ARENA using Equation 2.7 and 95% confidence level as a default value.
4. Output data was loaded into Excel and Equation 2.11 was used to calculate the minimum number of runs for the performance measures of interest, which were the output variables related to the utilization of personnel, machine or equipment up-time, throughput and work-in-process inventory. The following are the percentages used to calculate the desired half-width:
   - *Utilization of personnel* – it was decided to use 1% due to the cost associated with labor.
   - *Equipment up-time* – it was decided to use 1% due to the cost associated with equipment acquisition.
   - *Throughput* – it was decided to use 2.5% due to the risk associated with loosing sales. At an average of 821.4 units per month (average throughput from the "Actual System"), the company could loose $72k monthly if the average throughput of the real system comes 2.5% below the average

estimated by the simulation model. This could represent up to $865k per year if the minus 2.5% is consistent throughout the year.

- *Work-in-process inventory* – it was decided to use 10% since this represents only 2.2 units from the maximum allowable average inventory of 22 units.

5. The "Actual System" was modified by increasing the simulation run-length to 27,600 minutes, and the warm-up period to 9,200 minutes. The model was named as "Warm-up Period Validation". No changes were done to the number of runs as well as to the input parameters of the model. Please note that the simulation run-length was increased in proportion to the warm-up period, as a result, the same amount of minutes per replicate were used for data collection and output analysis as in the "Actual System". Data from all output variables of the model were recorded.
6. Step 2 was repeated for the "Warm-up Period Validation" model.
7. Step 3 was repeated for the "Warm-up Period Validation" model.
8. Step 4 was repeated for the "Warm-up Period Validation" model.
9. The Output Analyzer from ARENA was used to perform a hypothesis test as explained in section 2.4.1. The Output Analyzer used $\alpha = .05$ as a default value.
10. Finally, results from the hypothesis test were analyzed in order to validate or reject the warm-up period.

3.2.2. Sensitivity Analysis for Critical Input Parameters (Factors)

Sensitivity analyses were done with the main objective of finding out input parameters of the model that had no major impact on the performance measures of interest or to the objectives of the project. Once identified, those parameters were fixed to a particular value that allowed objectives to be reached, and were excluded from further experimental designs. In addition, another objective was to gain some knowledge of the behavior of the system in order to define the experimental levels for the input parameters considered as experimental factors.

In summary, four sensitivity analyses were done. The first analysis was a comparison between the "Actual System" and a new model, in which the inter-arrival rate was increased to receive more than 3000 units at the testing area. The objective was to confirm the criticality of the inter-arrival rate and to gain knowledge at how the "Actual System" would behave at 3000 units, which is the goal in terms of throughput. From this point on, the inter-arrival rate was setup for receiving at least 3000 units at test. The second sensitivity analysis was focused on the testing area. In general, each input parameter from the testing area was changed on a one by one basis. Based on the lessons learned at this stage, input parameters at testing were changed to allow 3000 units to reach the final inspection area (inspection I, draw out and final inspection) and the second sensitivity analysis was done changing only input parameters from the final inspection area. The same approach was repeated again, but this time setting input parameters at final inspection area to allow 3000 units reach the packing area, and then changing input parameters from the packing area only.

The methodology followed will be described in the next four sub-sections, following the order in which they were performed.

3.2.2.1."Actual System" vs. "3000 Units Through Test"

1. Table 3.1 was used to setup the input parameters for the models being compared. The simulation run-length was setup to 19,320 minutes, the warm-up period to 920 minutes, and the number of runs to 100. 18,400 minutes per replicate, were used for data collection and output analysis purposes. This is the equivalent to 920 minutes per day (2 shifts of 460 minutes each) on a month of 20 production days. Data from all of the output variables from both models were recorded. The first model was named as "Actual System" and the second as "3000 Units Through Test". Please note that the only change between both models is the inter-arrival time.

Table 3.1 Input Parameters – "Actual System" vs. "3000 Units Through Test".

	Testing Area	Actual System	3000 Units Through Test
Simulation Parameters	Inter-arrival time (Mins.)	Expo (14.3)	0.4*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	1	1
	Qty. testing personnel - Tester #2	1	1
	Qty. testing machines - Tester #1	1	1
	Qty. testing machines - Tester #2	1	1
	Tester#1 processing time	3+52*Beta(2.93,4.65)	3+52*Beta(2.93,4.65)
	Tester #2 processing time	3+52*Beta(2.93,4.65)	3+52*Beta(2.93,4.65)
	Yield % - Tester #1	72.6	72.6
	Yield % - Tester #2	72.6	72.6
	Draw Out Area	**Actual System**	**3000 Units Through Test**
	Qty. Assy. Personnel	1	1
	Qty. Final Insp. personnel	1	1
	Qty. Draw Out testing machines	1	1
	Packing Area	**Actual System**	**3000 Units Through Test**
	Qty. packing personnel	2	2

2. All performance measures except the "Throughput" were assumed to be independent and identically distributed without doing any statistical validation. This is a fair assumption based on the replicate/deletion method (please refer to section 2.3.4). As for the "Throughput", data was assumed to be independent as they are the result of different replicates, but a normality test was performed using Minitab to validate that they were identically distributed.
3. Confidence intervals for the simulation performance measures of both models were built by the ARENA-Output Analyzer option using Equation 2.7 and 95% confidence level.
4. Output data from both models were loaded into Excel and Equation 2.11 was used to calculate the minimum number of runs for the performance measures of interest, which were the output variables related to the utilization of

personnel, machine or equipment up-time, throughput and work-in-process inventory. The following are the percentages used to calculate the desired half-width:

- *Utilization of personnel* – it was decided to use 1% due to the cost associated with labor.
- *Equipment up-time* – it was decided to use 1% due to the cost associated with equipment acquisition.
- *Throughput* – it was decided to use 2.5% due to the risk associated with loosing sales. At an average of 824.1 units per month (average throughput from the "Actual System"), the company could loose $72k monthly if the average throughput of the real system comes 2.5% below the average estimated by the simulation model. This could represent up to $865k per year if the minus 2.5% is consistent throughout the year.
- *Work-in-process inventory* – it was decided to use 10% since this represents only 2.2 units from the maximum allowable average inventory of 22 units.

5. ARENA-Output Analyzer was used to perform a hypothesis test as explained in section 2.4.1. The Output Analyzer used $\alpha = .05$ as default value.
6. Finally, the results from the hypothesis test were analyzed in order to reject or fail to reject the hypothesis null (H_0).

3.2.2.2.Testing Area

1. Steps 1 thru 6 of section 3.2.2.1 were repeated for input parameters shown at Table 3.2. The models were named as “New System at 3000 Units Through Test” and “6 Testing Machines”. Please note that the only difference between both models is the number of testing machines.

Table 3.2 Input Parameters – “New System at 3000 Units Through Test” vs. “6 Testing Machines”.

	Testing Area	3000 Units Through Test	6 Testing Machines
Simulation Parameters	Inter-arrival time (Mins.)	0.4*(Expo (14.3))	0.4*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	2	2
	Qty. testing personnel - Tester #2		
	Qty. testing machines - Tester #1	2	6
	Qty. testing machines - Tester #2		
	Tester#1 processing time	3+52*Beta(2.93,4.65)	3+52*Beta(2.93,4.65)
	Tester #2 processing time		
	Yield % - Tester #1	72.6	72.6
	Yield % - Tester #2		
	Draw Out Area	**3000 Units Through Test**	**6 Testing Machines**
	Qty. Assy. personnel	1	1
	Qty. Final Insp. personnel	1	1
	Qty. Draw Out testing machines	1	1
	Packing Area	**3000 Units Through Test**	**6 Testing Machines**
	Qty. packing personnel	2	2

2. Steps 1 thru 6 of section 3.2.2.1 were repeated for input parameters shown at Table 3.3. The models were named as "New System at 3000 Units Through Test" and "90% Yield". Please note that the only difference between both models is the Yield.

Table 3.3 Input Parameters – "New System at 3000 Units Through Test" vs. "90% Yield".

	Testing Area	**3000 Units Through Test**	**90% Yield**
Simulation Parameters	Inter-arrival time (Mins.)	0.4*(Expo (14.3))	0.4*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	2	2
	Qty. testing personnel - Tester #2		
	Qty. testing machines - Tester #1	2	2
	Qty. testing machines - Tester #2		
	Tester#1 processing time	3+52*Beta(2.93,4.65)	3+52*Beta(2.93,4.65)
	Tester #2 processing time		
	Yield % - Tester #1	72.6	90.0
	Yield % - Tester #2		
	Draw Out Area	**3000 Units Through Test**	**90% Yield**
	Qty. Assy. personnel	1	1
	Qty. Final Insp. personnel	1	1
	Qty. Draw Out testing machines	1	1
	Packing Area	**3000 Units Through Test**	**90% Yield**
	Qty. packing personnel	2	2

3. Steps 1 thru 6 of section 3.2.2.1 were repeated for input parameters shown at Table 3.4. The models were named as "New System at 3000 Units Through Test" and "30% Reduction in Testing Process Time". Please note that the only difference between both models is the testing process time.

Table 3.4 Input Parameters – "New System at 3000 Units Through Test" vs. "30% Reduction Test. Proc. Time".

	Testing Area	3000 Units Through Test	30% Reduction Testing Processing Time
Simulation Parameters	Inter-arrival time (Mins.)	0.4*(Expo (14.3))	0.4*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	2	2
	Qty. testing personnel - Tester #2		
	Qty. testing machines - Tester #1	2	2
	Qty. testing machines - Tester #2		
	Tester#1 processing time	3+52*Beta(2.93,4.65)	0.7*(3+52*Beta(2.93,4.65))
	Tester #2 processing time		
	Yield % - Tester #1	72.6	72.6
	Yield % - Tester #2		
	Draw Out Area	**3000 Units Through Test**	**30% Reduction Testing Processing Time**
	Qty. Assy. personnel	1	1
	Qty. Final Insp. personnel	1	1
	Qty. Draw Out testing machines	1	1
	Packing Area	**3000 Units Through Test**	**30% Reduction Testing Processing Time**
	Qty. packing personnel	2	2

3.2.2.3.Final Inspection Area

1. Steps 1 thru 6 of section 3.2.2.1 were repeated for input parameters shown at Table 3.5. The models were named as "New System at 3000 Units Through Draw Out" and "4 Assemblers". Please note that the only difference between both models is the number of assemblers.

Table 3.5 Input Parameters – "New System at 3000 Units Through Draw Out" vs. "4 Assemblers".

	Testing Area	**3000 Units Through Draw Out**	**4 Assembly Personnel**
Simulation Parameters	Inter-arrival time (Mins.)	0.38*(Expo (14.3))	0.38*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	2	2
	Qty. testing personnel - Tester #2		
	Qty. testing machines - Tester #1	6	6
	Qty. testing machines - Tester #2		
	Tester#1 processing time	0.7*(3+52*Beta(2.93,4.65))	0.7*(3+52*Beta(2.93,4.65))
	Tester #2 processing time		
	Yield % - Tester #1	90.0	90.0
	Yield % - Tester #2		
	Draw Out Area	**3000 Units Through Draw Out**	**4 Assembly Personnel**
	Qty. Assy. personnel	1	4
	Qty. Final Insp. personnel	1	1
	Qty. Draw Out testing machines	1	1
	Packing Area	**3000 Units Through Draw Out**	**4 Assembly Personnel**
	Qty. packing personnel	2	2

2. Steps 1 thru 6 of section 3.2.2.1 were repeated for input parameters shown at Table 3.6. The models were named as "New System at 3000 Units Through Draw Out" and "4 Final Inspectors". Please note that the only difference between both models is the number of final inspectors.

Table 3.6 Input Parameters – "New System at 3000 Units Through Draw Out" vs. "4 Final Inspectors".

Simulation Parameters	**Testing Area**	**3000 Units Through Draw Out**	**4 Final Inspectors**
	Inter-arrival time (Mins.)	0.38*(Expo (14.3))	0.38*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	2	2
	Qty. testing personnel - Tester #2		
	Qty. testing machines - Tester #1	6	6
	Qty. testing machines - Tester #2		
	Tester#1 processing time	0.7*(3+52*Beta(2.93,4.65))	0.7*(3+52*Beta(2.93,4.65))
	Tester #2 processing time		
	Yield % - Tester #1	90.0	90.0
	Yield % - Tester #2		
	Draw Out Area	**3000 Units Through Test**	**4 Final Inspectors**
	Qty. Assy. personnel	1	1
	Qty. Final Insp. personnel	1	4
	Qty. Draw Out testing machines	1	1
	Packing Area	**3000 Units Through Test**	**4 Final Inspectors**
	Qty. packing personnel	2	2

3. Steps 1 thru 6 of section 3.2.2.1 were repeated for input parameters shown at Table 3.7. The models were named as "New System at 3000 Units Through Draw Out" and "3 Draw Out Machines". Please note that the only difference between both models is the number of draw out machines.

Table 3.7 Input Parameters – "New System at 3000 Units Through Draw Out" vs. "3 Draw Out Machines".

Simulation Parameters	**Testing Area**	**3000 Units Through Draw Out**	**3 Draw Out Machines**
	Inter-arrival time (Mins.)	0.38*(Expo (14.3))	0.38*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	2	2
	Qty. testing personnel - Tester #2		
	Qty. testing machines - Tester #1	6	6
	Qty. testing machines - Tester #2		
	Tester#1 processing time	0.7*(3+52*Beta(2.93,4.65))	0.7*(3+52*Beta(2.93,4.65))
	Tester #2 processing time		
	Yield % - Tester #1	90.0	90.0
	Yield % - Tester #2		
	Draw Out Area	**3000 Units Through Draw Out**	**3 Draw Out Machines**
	Qty. Assy. personnel	1	1
	Qty. Final Insp. personnel	1	1
	Qty. Draw Out testing machines	1	3
	Packing Area	**3000 Units Through Draw Out**	**3 Draw Out Machines**
	Qty. packing personnel	2	2

3.2.2.4.Packing Area

1. Steps 1 thru 6 of section 3.2.2.1 were repeated for input parameters shown at Table 3.8. The models were named as “New System at 3000 Units Through Packing” and “3 Packers”. Please note that the only difference between both models is the number of packers.

Table 3.8 Input Parameters – “New System at 3000 Units Through Packing” vs. “3 Packers”.

	Testing Area	3000 Units Through Packing	3 Packers
Simulation Parameters	Inter-arrival time (Mins.)	0.33*(Expo (14.3))	0.33*(Expo (14.3))
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	2	2
	Qty. testing personnel - Tester #2		
	Qty. testing machines - Tester #1	6	6
	Qty. testing machines - Tester #2		
	Tester#1 processing time	0.7*(3+52*Beta(2.93,4.65))	0.7*(3+52*Beta(2.93,4.65))
	Tester #2 processing time		
	Yield % - Tester #1	90.0	90.0
	Yield % - Tester #2		
	Draw Out Area	**3000 Units Through Packing**	**3 Packers**
	Qty. Assy. personnel	4	4
	Qty. Final Insp. personnel	4	4
	Qty. Draw Out testing machines	3	3
	Packing Area	**3000 Units Through Packing**	**3 Packers**
	Qty. packing personnel	2	3

3.3. Reducing the Problem with Experimental Designs

As previously mentioned, the main objective at this stage of the simulation study was to reduce the number of critical factors to be considered in the metamodels. This objective was achieved through a series of sequential experimental designs with major focus on the categorical and discrete input parameters, in particular, the labor balancing issue, the number of assemblers, the number of final inspectors and the number of draw out machines.

The expected result was to eliminate these factors from the critical list by fixing them to a level that would allow the model to achieve the objectives mentioned on section 1.5.1. In order to do so, 2^{k-p} fractional factorial design with center points and 2^k factorial design were used. The sensitivity analyses (refer to section 3.2 for the methodology and section 4.2 for the results) were the base for setting the levels for each factor considered on the following experiments. It is important to mention that at this stage, the simulation was set to run the equivalent of 1 shift and to achieve the goal of 1000 units per month per shift. This decision was based on the conclusions of the sensitivity analyses (refer to section 4.2.2.5).

The section has been divided in two sub-sections: (1) Labor Balancing, and (2) Setting Up the Draw Out/Final Inspection Area. The methodology followed is described ahead.

3.3.1. Labor Balancing

It was observed from the execution of the sensitivity analysis (refer to section 4.2.2.1 of this report) a labor balancing (work load) problem related to locating the task Inspection I (refer to Figure 1.1) at the testing area vs. locating it at the draw out/final inspection area. It was decided to run a 2^{8-2} fractional factorial design with 10 center points and 1 replicate In order to understand the behavior of the model with respect to the labor balancing issue. This is a fractional factorial design of resolution V with a total of 74 experimental runs. Center points were added with the objective of having a better understanding on the behavior of the performance measures (or responses) with respect to

the presence of curvature. Only one replicate was specified in order to keep the number of design points at the lowest number that was thought could give the necessary information to decide about the balancing issue.

Statistically speaking, one replicate was of concern because at least two replicates were needed to have an estimate of the error. So the error obtained with this experiment was not a true estimate of the error itself. For this reason, it was decided to focus the analysis on good graphical plots rather than in analysis of variance. This approach made sense since the focus was on the categorical parameter and helped on keeping the simulation run time on the minimum possible.

The following was the methodology:

1. Minitab[12] was used to design the 2^{8-2} fractional factorial design. The following was specified in order for Minitab to generate the design:
   - 10 center points and 8 factors; 2 levels per factor (refer to Table 3.9).
   - Factors were coded (-1, 0, 1).

Table 3.9 2^{8-2} fractional factorial design with center point - Factor levels.

Performance Measure	Factor	Coded Levels		
		-1	0	1
Inter-arrival Time	A	0.8	0.6	0.4
Number of Testing Machines	B	3	4	5
Number of Testing Personnel	C	2	3	4
Yield (%)	D	72.6	81.3	90.0
Testing Process Time (%)	E	100.0	85.0	70.0
Number of Assembly Personnel	F	2	3	4
Number of Final Inspectors	G	1	3	5
Location of Inspection I	H	Draw Out	N/A	Testing

Note: Categorical factors do not have center point.

[12] Minitab – is a software package for statistical applications.

- ¼ fraction of 8 factors was selected for the design. This selection gave an experiment design of resolution V.
- 1 replicate and 1 block.

Refer to Appendix D (output from Minitab) for more detail on the design. Figure 3.1 is a graphical plot of the experimental design. Note that the number of draw out machines was not included in this experiment as a critical factor. This was based on the results of the sensitivity analyses (refer to section 4.2.2.5).

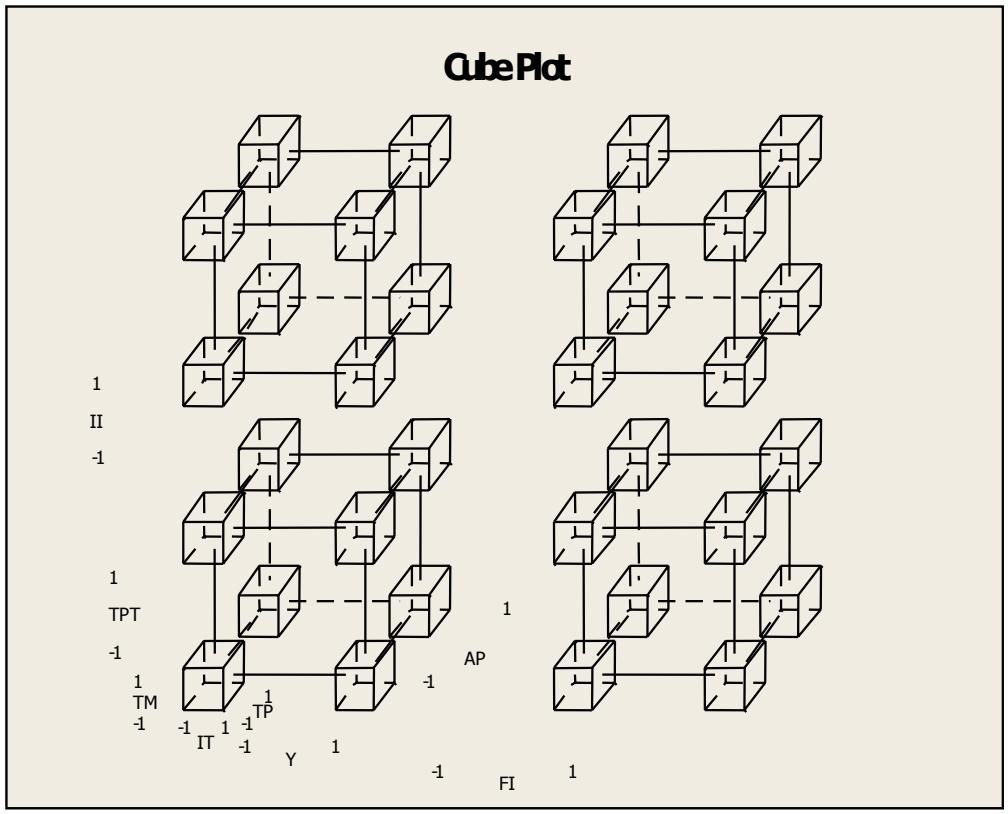

Figure 3.1 Cube Plot of the 2^{8-2} fractional factorial design with center points.

2. Each experimental design point was run with ARENA, setting the simulation run-length to 18,400 minutes, the warm-up period to 9,200 minutes, and the number of runs to 100. 9,200 minutes per replicate were used for data collection and output analysis purposes. This was the equivalent to 460 minutes per day (1 shift) on a month of 20 production days. In addition, the

number of resources related to dielectric machines, draw out machines and packing personnel were set to 1, 2, and 2, respectively. Since the experimental design had 10 replications at the center point, simulation seeds for the random generators related to the experimental factors were changed with every replicate. Each design point was filed with a different name, and data from all output variables were recorded for future reference if needed.

3. The average output for each performance measure of interest, except for the throughput, was entered into Minitab for analysis purposes. In the case of the throughput, it was decided to use the minimum value observed from the simulation runs in an effort to offer the end user with a system that will achieve the expected throughput most of the time, instead of 50% of the time which is what the average would do.
4. The labor cost per unit was calculated and entered into Minitab as a performance metric. The calculation was done by estimating the labor cost for each design point using Table 1.2, and dividing by the throughput data from the respective design point. Equation 3.1 was used for the calculation of the labor cost per unit for a particular design point:

$$LaborCost/Unit = \frac{\sum_{i=1}^{3}(\#emp_i)(\$/hr - emp_i)(40)(52)}{(12)(Throughput)} \tag{3.1}$$

where, i is the personnel classification (testing, assembly or final inspection), $\#emp_i$ is the number of employee with that classification, $\$/hr - emp_i$ is the hourly cost of an employee with classification i, 40 is the total number of regular hours in a week, 52 is the total number of weeks in a year, 12 is the total number of months in a year, and *Throughput* is the throughput obtained from the simulation for that particular design point. Note that cost related to packing personnel was not included in the equation because the number of packing personnel was a fix value.

5. Each response variable was analyzed using Minitab. As previously mentioned, the focus of the analysis was on the labor balancing issue and graphical plots

were the main resources used at this stage of the analysis. The final decision to resolve the balancing issue was taken after analyzing the labor cost per unit ($/unit) performance metric.

3.3.2. Setting Up the Draw Out/Final Inspection Area

In order to understand the behavior of the model with respect to the draw out/final inspection area, it was decided to run a 2^3 full factorial design with three replicates and three blocks. The design had a total of 24 experimental runs. Three replicates were specified in order to estimate the error. The simulation seeds for the random generators related to the experimental factors were changed with every replicate. Blocking was done for each new replicate in order to have a better understanding about the impact that the seeds could have on the statistical analysis of the experiment.

The main objective of this experimental design was to define the fix values at which to set the number of assembly personnel, the number of final inspection personnel and the number of draw out machines. Keep in mind that for the monthly throughput, the objective was 1000 units in one shift (equivalent to 3000 in 3 shifts); all other objectives remained the same (refer to section 1.5.1). All input parameters from the testing area were fixed to a value that allowed the model to process a total of 1000 units monthly at the packing station in each design point.

The decision with respect to the value at which to fix each factor was based on the labor cost per unit, the labor utilization for the assembly and the final inspection personnel, the up-time for the draw out machine, and the work-in-process inventory, in particular, the buffer in front of the draw out machine. The approach was to define the combination of assembly personnel, final inspection personnel, and draw out machines that minimized the labor cost per unit and met the constraints related to labor utilization, machine up-time and work-in-process inventory.

As in the previous experimental design, the analysis relied mostly on graphical plots. This approach made sense since the focus was on discrete input parameters. The following was the methodology:

1. Minitab was used to design the 2^3 factorial design. The following was specified in order for Minitab to generate the design:
   - 3 factors; 2 levels per factor (refer to Table 3.10).
   - Factors were coded (-1, 1).
   - 3 replicates.
   - 3 blocks.

Table 3.10 2^3 factorial design - Factor levels.

Performance Measure	Factor	Coded Levels	
		-1	1
Number of Assembly Personnel	A	3	4
Number of Final Inspectors	B	2	3
Number of D.O. Machines	C	2	3

Refer to Appendix E (output from Minitab) for more detail on the design. Figure 3.2 is a graphical plot of the experimental design.

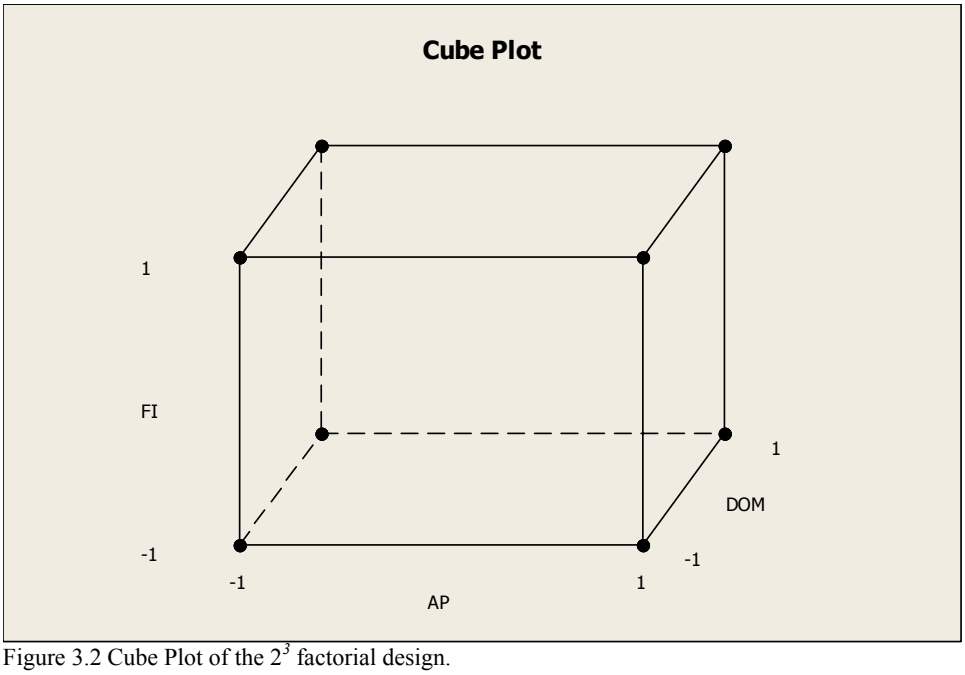

Figure 3.2 Cube Plot of the 2^3 factorial design.

2. Each experimental design point was run with ARENA, setting the simulation run-length to 18,400 minutes, the warm-up period to 9,200 minutes, and the number of runs to 100. 9,200 minutes per replicate were used for data collection and output analysis purposes. This was the equivalent to 460 minutes per day (1 shift) on a month of 20 production days. The number of resources related to dielectric machines, testing machines, testing personnel and packing personnel were set to 1, 5, 4, and 2, respectively. The inter-arrival time and the testing process time were set to 46.5% and 70% of the values observed from the real system. In addition, the yield was set to 90%. Since the experimental design had 3 replications, simulation seeds for the random generators related to the experimental factors were changed with every replicate. Each design point was filed with a different name, and data from all output variables were recorded for future reference if needed.
3. The average output for each performance measure of interest, except for the throughput, was entered into Minitab for analysis purposes. In the case of the throughput, it was decided to use the minimum value observed from the simulation runs in an effort to offer the end user with a system that will achieve the expected throughput most of the time, instead of 50% of the time which is what the average would do.
4. The labor cost per unit was calculated and entered into Minitab as a performance metric. The calculation was done by estimating the labor cost for each design point using Table 1.2, and dividing by the throughput data from the respective design point. Equation 3.2 was used for the calculation of the labor cost per unit for a particular design point:

$$LaborCost/Unit = \frac{\sum_{i=1}^{2}(\#emp_i)(\$/hr - emp_i)(40)(52)}{(12)(Throughput)} \tag{3.2}$$

where, i is the personnel classification (assembly or final inspection), $\#emp_i$ is the number of employees with that classification, $\$/hr - emp_i$ is the hourly cost of an employee with classification i, 40 is the total number of regular

hours in a week, 52 is the total number of weeks in a year, 12 is the total number of months in a year, and *Throughput* is the throughput obtained from the simulation for that particular design point. Note that cost related to testing and packing personnel was not included in the equation because the number of testing personnel and packing personnel were fixed values.

5. Each response variable of interest was analyzed using Minitab.

3.4. Development of Metamodels

This section describes a series of parallel experimental designs that were performed to develop the metamodels used in the optimization process of the study. These experimental designs focused on the testing area since the input parameters from the draw out/final inspection and packing areas were already fixed to a particular set of values; thus, reducing the complexity of the model.

The testing area had four critical factors: inter-arrival time, number of testing machines, yield and testing process time. All of the four critical factors were quantitative input parameters, but the number of testing machines was discrete. In addition, it was already understood at this stage of the simulation study that some of the performance measures of interest could have a quadratic relationship with these factors. It was decided to run three parallel central composite experimental designs; one experiment for each testing machine.

The evaluation of the outcomes from the experimental designs was based on statistical techniques generally used for these purposes, such as: hypothesis tests, analysis of variance, normality test, run charts, surface plots and a series of other commonly used graphical plots. The expected result from this stage of the simulation study was a set of algebraic expressions for each experiment that expressed the relationship between the factors (input parameters) and the response variables of interest.

The methodology followed is described in the next three sub-sections: (1) Metamodels for 3 Testing Machines, (2) Metamodels for 4 Testing Machines, and (3)

Metamodels for 5 Testing Machines. Whenever appropriate, the section refers to the Appendixes for backup information or data.

3.4.1. Metamodels for 3 Testing Machines

This section describes the experiment performed to develop the metamodels related to the behavior of the simulation model when three testing machines were used. The experiment was a central composite design with three factors with two levels each, two replicates and two blocks. A total of 40 experimental runs were done for each experiment: 16 for cube points, 12 for center points in cube, and 12 for axial points. The value for α, distance of the axial runs from the design center (refer to section 2.5.4), was 1.68179, which was the default value from Minitab for a central composite design with those specifications.

The performance measures of interest were the throughput, the up-time for the testing machines, and the work-in-process inventory. For each of these response variables, regression analysis was done with the respective analysis of variance including the lack of fit test. In addition, graphical plots for corroboration of the assumptions of the analysis of variance were done as well as surface plots.

The methodology followed is explained next:

1. Minitab was used to design the 2^3 central composite factorial design. The following was specified in order for Minitab to generate the design:
   - 3 factors; 2 levels per factor with center point and axial points.
   - Factors were coded (-1.68179, -1, 0, 1, 1.68179). Refer to Table 3.11.

Table 3.11 2^3 central composite factorial design - Factor levels.

Performance Measure	Factor	Coded Levels				
		-1.68179	-1	0	1	1.68179
Inter-arrival Time	A	0.485000	0.480946	0.475000	0.469054	0.465000
Yield (%)	B	0.850000	0.860135	0.875000	0.889865	0.900000
Testing Process Time (%)	C	0.750000	0.739865	0.725000	0.710135	0.700000

- 2 replicates
- 2 blocks.

Refer to Appendix F (output from Minitab) for more detail on the design.

2. Each experimental design point was run with ARENA, setting the simulation run-length to 18,400 minutes, the warm-up period to 9,200 minutes, and the number of runs to 100. 9,200 minutes per replicate were used for data collection and output analysis purposes. This was the equivalent to 460 minutes per day (1 shift) on a month of 20 production days. The number of resources related to dielectric machines, testing personnel, assembly personnel, final inspection personnel, number of draw out machines, and packing personnel were set to 1, 2, 4, 2, 3, and 2, respectively. The inter-arrival time, the testing process time, and yield were the experimental factors. The simulation seeds for the random generators related to the experimental factors were changed with every replicate of a design point. Each design point was filed with a different name, and data from all output variables were recorded for future reference if needed.
3. The average output for the performance measures of interest was entered into Minitab for analysis purposes.
4. Each response variable of interest was analyzed using Minitab. The following analysis was done: regression, analysis of variance with lack of fit test, normal probability plot for residuals, histogram of the residuals, plot of the residual vs. fitted values, plot of the residuals vs. the order of the data.

3.4.2. Metamodels for 4 Testing Machines

This section describes the experiment performed to develop the metamodels related to the behavior of the simulation model when four testing machines were used. The experiment was a central composite design with three factors with two levels each, two replicates and two blocks. A total of 40 experimental runs were done for each experiment: 16 for cube points, 12 for center points in cube, and 12 for axial points. The

value for α, distance of the axial runs from the design center (refer to section 2.5.4), was 1.68179, which was a default value from Minitab for a central composite design with those specifications.

The performance measures of interest were the throughput, the up-time for the testing machines, and the work-in-process inventory. For each of these response variables, regression analysis was done with the respective analysis of variance including the lack of fit test. In addition, graphical plots for corroboration of the assumptions of the analysis of variance were done as well as surface plots.

The methodology followed is explained next:

1. Minitab was used to design the 2^3 central composite factorial design. The following was specified in order for Minitab to generate the design:

   - 3 factors; 2 levels per factor with center point and axial points.
   - Factors were coded (-1.68179, -1, 0, 1, 1.68179). Refer to Table 3.12.

Table 3.12 2^3 central composite factorial design - Factor levels.

Performance Measure	Factor	Coded Levels				
		-1.68179	-1	0	1	1.68179
Inter-arrival Time	A	0.485000	0.477905	0.467500	0.457094	0.450000
Yield (%)	B	0.800000	0.820270	0.850000	0.879730	0.900000
Testing Process Time (%)	C	1.000000	0.939190	0.850000	0.760810	0.700000

   - 2 replicates.
   - 2 blocks.

   Refer to Appendix F (output from Minitab) for more detail on the design.
2. Each experimental design point was run with ARENA, setting the simulation run-length to 18,400 minutes, the warm-up period to 9,200 minutes, and the number of runs to 100. 9,200 minutes per replicate were used for data collection and output analysis purposes. This was the equivalent to 460 minutes per day (1 shift) on a month of 20 production days. The number of

resources related to dielectric machines, testing personnel, assembly personnel, final inspection personnel, number of draw out machines, and packing personnel were set to 1, 2, 4, 2, 3, and 2, respectively. The inter-arrival time, the testing process time, and yield were the experimental factors. The simulation seeds for the random generators related to the experimental factors were changed with every replicate of a design point. Each design point was filed with a different name, and data from all output variables were recorded for future reference if needed.

3. The average output for the performance measures of interest was entered into Minitab for analysis purposes.
4. Each response variable of interest was analyzed using Minitab. The following analysis was done: regression, analysis of variance with lack of fit test, normal probability plot for residuals, histogram of the residuals, plot of the residual vs. fitted values, plot of the residuals vs. the order of the data.

3.4.3. Metamodels for 5 Testing Machines

This section describes the experiment performed to develop the metamodels related to the behavior of the simulation model when five testing machines were used. The experiment was a central composite design with three factors with two levels each, two replicates and two blocks. A total of 40 experimental runs were done for each experiment: 16 for cube points, 12 for center points in cube, and 12 for axial points. The value for α, distance of the axial runs from the design center (refer to section 2.5.4), was 1.68179, which was a default value from Minitab for a central composite design with those specifications.

The performance measures of interest were the throughput, the utilization for the packing personnel, and the work-in-process inventory. For each of these response variables, regression analysis was done with the respective analysis of variance including the lack of fit test. In addition, graphical plots for corroboration of the assumptions of the analysis of variance were done as well as surface plots.

The methodology followed is explained next:

1. Minitab was used to design the 2^3 central composite factorial design. The following was specified in order for Minitab to generate the design:

   - 3 factors; 2 levels per factor with center point and axial points.
   - Factors were coded (-1.68179, -1, 0, 1, 1.68179). Refer to Table 3.13.

   Table 3.13 2^3 central composite factorial design - Factor levels.

Performance Measure	Factor	Coded Levels				
		-1.68179	**-1**	**0**	**1**	**1.68179**
Inter-arrival Time	A	0.4850000	0.4687840	0.4450000	0.4212160	0.4050000
Yield (%)	B	0.7260000	0.7612698	0.8130000	0.8647302	0.9000000
Testing Process Time (%)	C	1.0000000	0.9391900	0.8500000	0.7608100	0.7000000

   - 2 replicates.
   - 2 blocks.

   Refer to Appendix F (output from Minitab) for more detail on the design.

2. Each experimental design point was run with ARENA, setting the simulation run-length to 18,400 minutes, the warm-up period to 9,200 minutes, and the number of runs to 100. 9,200 minutes per replicate were used for data collection and output analysis purposes. This was the equivalent to 460 minutes per day (1 shift) on a month of 20 production days. The number of resources related to dielectric machines, testing personnel, assembly personnel, final inspection personnel, number of draw out machines, and packing personnel were set to 1, 2, 4, 2, 3, and 2, respectively. The inter-arrival time, the testing process time, and yield were the experimental factors. The simulation seeds for the random generators related to the experimental factors were changed with every replicate of a design point. Each design point was filed with a different name, and data from all output variables were recorded for future reference if needed.

3. The average output for the performance measures of interest was entered into Minitab for analysis purposes.
4. Each response variable of interest was analyzed using Minitab. The following analysis was done: regression, analysis of variance with lack of fit test, normal probability plot for residuals, histogram of the residuals, plot of the residual vs. fitted values, plot of the residuals vs. the order of the data.

3.5. Optimization Tool

This section describes the optimization tool developed for the practical application of this project. The main objective of the tool was to determine the number of machines that were required to meet the expected production as well as the constraints specified by the management of the company, given a combination of the input parameters yield and testing process time.

It would have been easier for the analysis and execution of this project, fixing the yield and testing process time to their respective optimum level. Nevertheless, this would have not been practical for the end user due to the uncertainty with respect to achieving those levels. Basically, the objective was to develop a tool that could be used for management decision making. This in fact was the complexity behind this project, since we have tried so far to understand the behavior of the system for the whole spectrum of alternatives or possibilities with respect to the yield and testing process time.

Typical simulation optimization techniques such as the ones discussed in section 2.6, were not practical options for the end user in terms of the availability of resources, contrary to developing a set of metamodels that could easily be used in exchange of the simulation model. The metamodels were used to build mathematical programming models for optimization purposes. These models were of the type of smooth non-linear optimization problems with 3 decision variables and 3 constraints, where at least one restriction was not linear. Microsoft Excel Solver provided the desired easiness of use and represented an inexpensive alternative for optimization purposes since it was already

available for the end user. In addition, it was mentioned on section 2.7 that the Generalized Reduced Gradient method employs by Microsoft Excel Solver could solve this type of problem for up to 200 decision variables and 100 constraints.

The tool was developed using Microsoft Excel Solver and Visual Basic for Applications. Basically, the end user enters the values for the yield and testing process time, the tool selects the optimization model (for 3, 4 or 5 testing machines) and Microsoft Excel Solver finds the optimum based on maximizing an objective function and meeting the restrictions imposed by the metamodels. The search variable is the inter-arrival time. It is important to mention that the search is done within the experimental region, thus the experimental region was critical for finding the global optimum.

This section has been divided in two sub-sections: (1) Description of the Optimization Tool, and (2) Validation of the Optimization Tool.

3.5.1. Description of the Optimization Tool

Even though there was no contact with the company representatives at this stage of the project, the optimization tool was developed having the end user in mind. It was decided to use Microsoft Excel Solver and Visual Basic for Applications due to their universality in today's business world.

Figure 3.3 shows the data entry screen. The end user enters the values for the yield and testing process time and clicks on "Next>>". Visual Basic for Applications revises that the values are valid numbers and select the optimization model to be used (refer to Appendix G). Figure 3.4 shows the Excel worksheet for optimization purposes. There is one worksheet for each optimization model (3, 4 or 5 testing machines). The end user enters information with respect to the sales price per unit, inventory cost per unit, acquisition cost per the testing machine, book value at the end of the depreciation period, and the depreciation period. The end user selects the solver option from Excel and the optimization is run.

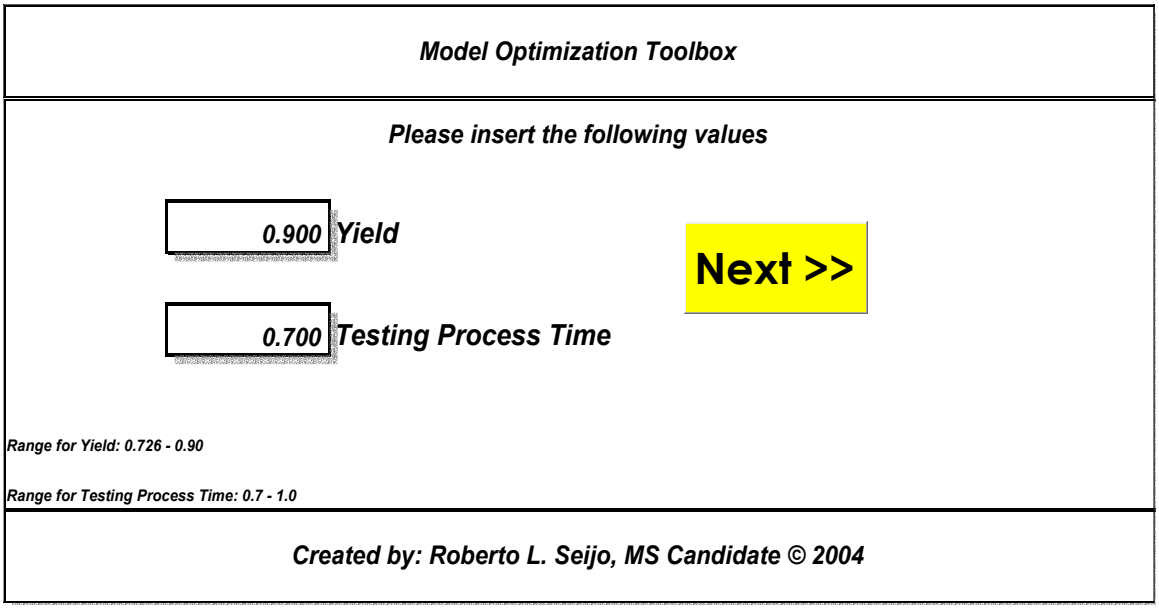

Figure 3.3 Data entry screen.

Optimization w/ 3 Testing Machines					
Interarrival Time (IT)	Coded	1.682	Metamodel		Limits
Yield (Y)	Coded	1.682	Up-Time Testing Machine	94.3%	95.0%
	Uncoded	90.0%	WIP	18.6	22
Testing Process Time (TPT)	Coded	1.682	Throughput	1090.4	1000
	Uncoded	70.0%	Max. Income (Objective)	$3,787,773.33	

STEPS:		
1. Enter the sales price/unit.	$ 3,500.00	6. Select "Tools" menu.
2. Enter the inventory cost/unit.	$ 1,500.00	7. Select "Solver" option.
3. Enter the equipment acquisition cost.	$ 350,000.00	8. Select "Solver" tab.
4. Enter the book value at the end of depreciation period.	$ 50,000.00	
5. Enter depreciation period in terms of years.	10	

Figure 3.4 Optimization Screen.

The following are the objective functions used for the optimization model for each testing machine (3, 4, or 5). Z is the monthly cost function per shift of the

optimization model, which represents the income generated by the throughput less the cost related to the work-in-process and the depreciation of the testing machines.

- *3 Testing Machines*

$$Max\ Z = (SP)(THPT) - (IC)(WIP) - ((((EAC - BV)/DY)/12)/3 \quad (3.3)$$

Where *SP* is the sales price per unit, *THPT* is the throughput, *IC* is the inventory or material cost per unit, *WIP* is the work-in-process, *EAC* is the acquisition cost for a new testing machine, *BV* is the book value at the end of the depreciation period, *DY* is the depreciation period in years, 12 is the number of months in a year, and 3 is the number of shifts. Equation (4.1), Equation (4.2), and Equation (4.3) were used to build the restrictions of the model.

- *4 Testing Machines*

$$Max\ Z = (SP)(THPT) - (IC)(WIP) - (((((EAC - BV)/DY)/12)/3)*2 \quad (3.4)$$

Where *SP* is the sales price per unit, *THPT* is the throughput, *IC* is the inventory or material cost per unit, *WIP* is the work-in-process, *EAC* is the acquisition cost for a new testing machine, *BV* is the book value at the end of the depreciation period, *DY* is the depreciation period in years, 12 is the number of months in a year, 3 is the number of shifts, and 2 is the number of new testing machines. Note that the portion of the equation related to depreciation is multiplied by 2; the reason for this is that at the present, there are 2 testing machines being used at the company. Equation (4.4), Equation (4.5), and Equation (4.6) were used to build the restrictions of the model.

- *5 Testing Machines*

$$Max\ Z = (SP)(THPT) - (IC)(WIP) - ((((EAC - BV)/DY)/12)/3)*3 \qquad (3.5)$$

Where *SP* is the sales price per unit, *THPT* is the throughput, *IC* is the inventory or material cost per unit, *WIP* is the work-in-process, *EAC* is the acquisition cost for a new testing machine, *BV* is the book value at the end of the depreciation period, *DY* is the depreciation period in years, 12 is the number of months in a year, 3 is the number of shifts, and 3 is the number of new testing machines. Note that the portion of the equation related to depreciation is multiplied by 3; the reason for this is that at the present, there are 2 testing machines being used at the company. Equation (4.7), Equation (4.8), and Equation (4.9) were used to build the restrictions of the model.

3.5.2. Validation of the Optimization Tool

The validation was done mostly by focusing on three subjects: (1) predicting the accuracy of the metamodels vs. the simulation output, (2) the ability of the experimental region to contain a local optimum with respect the combination of possible values for the yield and testing process time input parameters, and (3) the ability of the experimental region to contain the global optimum.

For subject (1), it was decided to run three scenarios for possible combinations of yield and testing process time (refer to Table 3.14). The output from the optimization tool was then compared to the output of the simulation model. It is important to mention that for simulation purposes, the input parameter inter-arrival time was obtained from the optimization tool.

Table 3.14 Validation Scenarios for the metamodels accuracy.

	Input		Output
Scenario #	Yield (%)	Testing Proc. Time (%)	Optimization Model
1	90	70	3 Testing Machines
2	81.3	85	4 Testing Machines
3	72.6	100	5 Testing Machines

It is common to validate metamodels by means of confidence interval hypothesis tests. In this approach, the confidence interval for the prediction is statistically compared to the confidence interval of the simulation response. It was understood that for the case of this project, such a comparison was of no practical value for the end user or management of the company. The main reason being that the management was more interested in knowing the probability distribution for the performance measure throughput, rather than the confidence interval for the mean (average throughput) at the stationary point. In other words, they were more interested in the behavior of the population at the stationary point. It was then decided to do the validation by comparing the point estimate of the simulation for the responses of interest (throughput, up-time testing machines, utilization packing personnel and work-in process inventory) vs. the prediction from the metamodels. The decision criteria for validation purposes were the same percentages used for estimating the number of simulation replicates at the time of doing the sensitivity analysis (refer to section 3.2.1). The criteria were of practical value since they represented the allowable error for each performance measure given by the management of the company. It was understood that in the eventuality of presenting these results to the management, this validation approach was of common sense for them, thus gaining confidence on the results of the project.

In order to calculate the point estimate for the performance measures or simulation responses of interest, 1000 replicates were run for each validation scenario. Since 1 replicate was equivalent to a month worth of production, then 1000 replicates were equivalent to 83.3 years; from the point of view of the manufacturing life for this product, such amount of replicates were understood to be representative of the

population. Statistically speaking, the confidence interval for each simulation response of interest was small enough to consider the simulation averages as the point estimates at the steady state.

For subject (2), the optimization tool was run with two scenarios presented in Table 3.15. Scenario #1 represented the limit for the 3 testing machines experimental region (in terms of yield and testing process time). Scenario #2 was the limit for 4 testing machines. In the case of scenario #1, the expectation was for the optimization tool not to find a feasible solution with the optimization modeling for 3 testing machines, but with the optimization modeling for 4 testing machines. This guaranteed that for any combination of yield and testing process time feasible for 3 testing machines, the optimum was contained within the experimental region, at least within the limits provided for yield and testing process time. The same being true for scenario #2 and 4 testing machines, but in this case, the feasible solution lying within 5 testing machines optimization model. This approach was feasible because the combination of 90% yield, 70% of the testing process time, and 48.5% of the inter-arrival time, was a common point or the starting point for the development of the three experimental regions. In other words, the region for 3 testing machines was contained within the region for 4 testing machines, which in turn was contained within the region for 5 testing machines.

Table 3.15 Validation Scenarios for the experimental regions.

	Input		Output	
Scenario #	Yield (%)	Testing Proc. Time (%)	No Feasible Solution	Feasible Solution
1	85	75	3 Testing Machines	4 Testing Machines
2	81.3	85	4 Testing Machines	5 testing machines

Finally, for subject (3), the conclusion was made by observation of the results from the previous 5 validation scenarios. In order for the experimental region to contain the global optimum for any combination of yield and testing process time, the inter-arrival time for the feasible solution must not be the experimental limits defined by ±1.68179. Nevertheless, and in the eventuality that the global optimum was not within

the region defined by the inter-arrival time, the decision with respect to the number of testing machines for a particular combination of yield and testing process time were still valid as long as the previous two subjects were validated.

3.6. Summary

This chapter described the methodology followed for the execution of this project. It covered the use of sensitivity analysis, experimental designs, regression analysis for building metamodels, and the description of the optimization tool developed as well as its validation process.

Several sensitivity analyses were defined at the early stage of this project with two main objectives: (1) to validate the warm-up period of the simulation model for the actual system, and (2) to determine the simulation input parameters that should be considered as experimental factors. Validating the number of runs, and performing hypothesis tests for two systems comparison were must do steps as part of the methodology described.

This chapter described the methodology followed for the use of 2^k factorial designs and 2^{k-p} fractional factorial designs. The main objective of these experiments was to minimize the complexity of the initial model by finding levels at which to set non-critical input parameters. The use of central composite designs with the objective of developing metamodels for the performance measures of interest was also explained in this chapter.

The description of the optimization tool developed with Microsoft's Excel Solver and Visual Basic for Applications was described in this chapter. In addition, the methodology followed for the validation of the tool was also described. It was mentioned that the mathematical programming model was of the smooth non-linear type which Microsoft Excel Solver can find a solution by means of the Generalized Reduced Gradient algorithms

The following chapter (IV) will discuss the results and statistical analysis for the sensitivity runs, experimental designs, and the optimization tool validation process. Chapter V will present the conclusions, and recommendations in terms of future research.

CHAPTER IV. Analysis and Results

4.1. Introduction

This section explains the results obtained from the application of the methodology described in the previous chapter. The section has been divided into five sub-sections: (1) Sensitivity analysis (2) Reducing the problem with experimental designs, (3) Development of metamodels, (4) Optimization tool and (5) Comparison of methodologies.

Section (1) explains the results obtained from the sensitivity analyses. Section (2) explains the results obtained from the sequential experimental designs executed in order to reduce the number of critical factors to be considered in the metamodels. Section (3) explains the results obtained from parallel experimental designs executed in order to develop the metamodels. Section (4) describes the validation for the optimization tool. Finally, section (5) presents the comparison between the results obtained from this project and the results obtained from the previous project. It is important to remember that the previous project pretended to solve the practical problem by trial and error approach.

The results are presented by means of tables, charts and/or text. Each subsection covers the explanation and conclusions with respect to the results. Whenever appropriate, the section refers to the Appendixes for backup information or data.

4.2. Sensitivity Analysis

This section explains the results and conclusions related to the sensitivity analyses. As previously mentioned, sensitivity analyses were done with respect to two main objectives: (1) to validate the warm-up period defined during the simulation analysis of the model for the actual (real) system, and (2) to determine the simulation input parameters that should be considered as experimental factors and their respective experimental levels. The methodology followed was pretty much the same for both cases. In general, the objective was to vary one factor at a time to determine its impact on the

responses of the simulation. Any input parameter that statistically impacted by means of hypothesis testing at least one performance measure was considered as an experimental factor. For more details on the methodology, please refer to sections 3.2.1 and 3.2.2.

4.2.1. Validation of the Warm-up Period

The output analysis started with the normality test for the performance measure "Throughput", for which the assumption of independency was valid but not necessarily the assumption of being identically distributed. All other variables were assumed to be independent and identically distributed.

The assumption of normality for the simulation output variable "Throughput" for the "Actual System" was validated by the normality test (refer to Figure 4.1). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

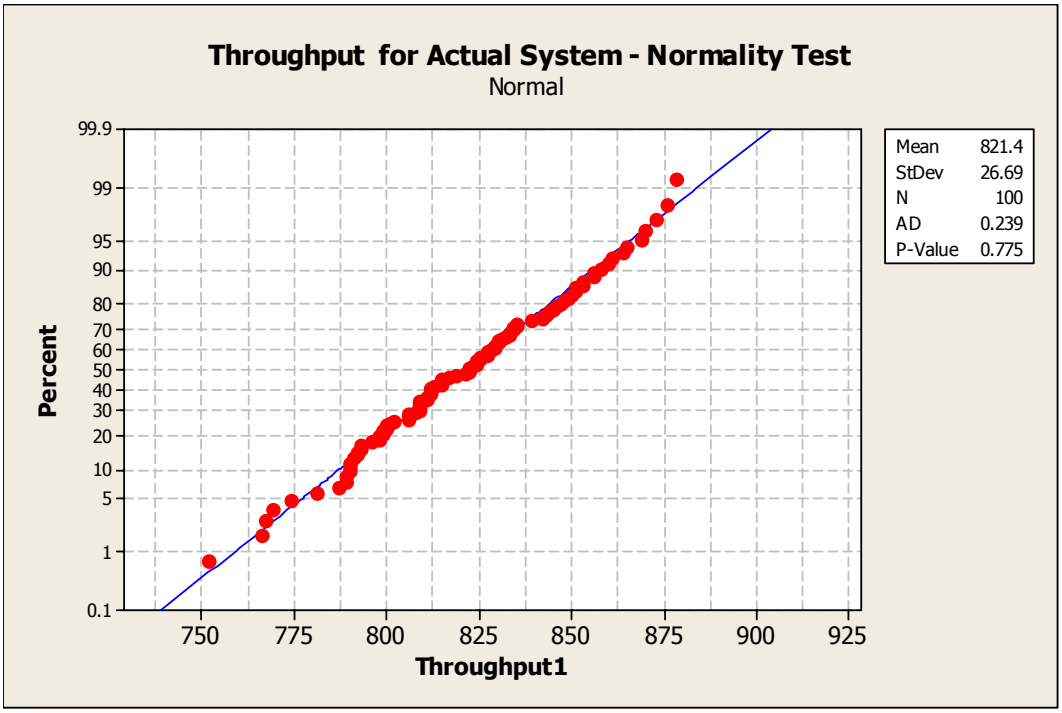

Figure 4.1 Normality test – "Actual System" model.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

The assumption of normality for the simulation output variable "Throughput" for the "Warm-up Period Validation" was validated by the normality test (refer to Figure 4.2). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

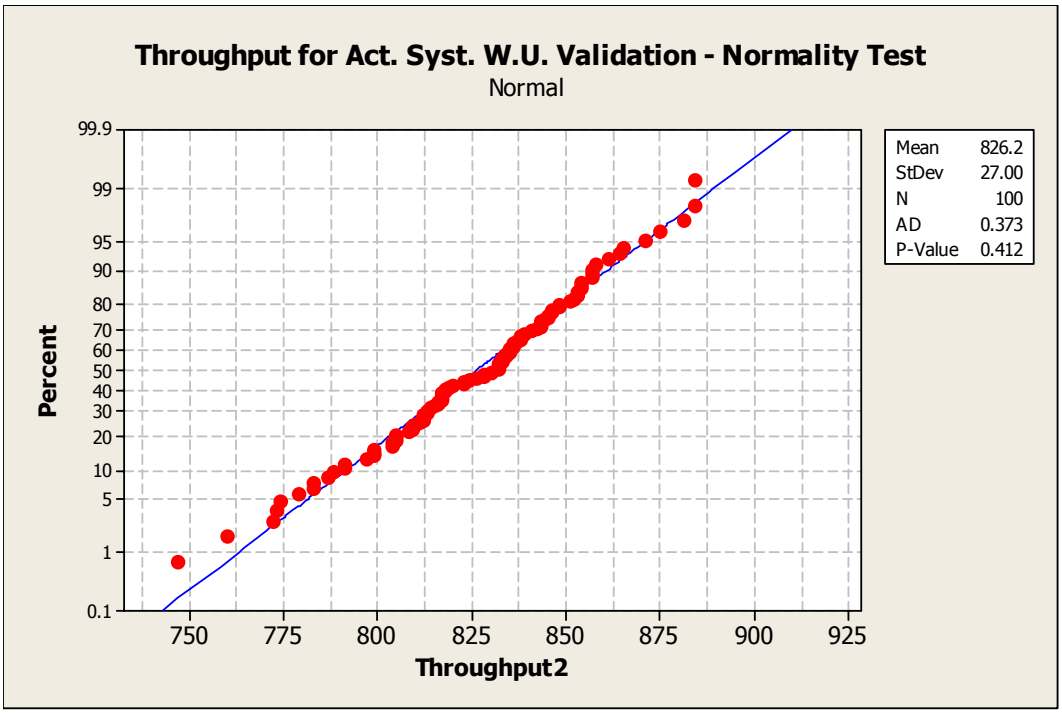

Figure 4.2 Normality test – "Warm-up Period Validation" model.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

Table 4.1 shows the simulation responses or output variables of interest for the "Actual System" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, and the confidence interval (minimum

and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. It was concluded from this table that no additional replicates were required.

Table 4.1 Validation of the warm-up period – "Actual System" model.

	Actual System						
Testing Area	**Average $\mu 1$**	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	10.5%	9.8%	11.4%	10.5%	10.6%	100.0	38.1
% Up-time Test. Mach. - Tester #1	90.0%	82.2%	97.0%	89.4%	90.6%	100.0	45.1
% Up-time Test. Mach. - Tester #2	90.2%	82.2%	97.7%	89.6%	90.8%	100.0	46.9
% Util. Test. Pers. - Tester#1	18.4%	17.1%	19.6%	18.3%	18.5%	100.0	31.5
% Util. Test. Pers. - Tester#2	18.4%	17.0%	19.8%	18.3%	18.5%	100.0	37.1
% Up-time Draw Out tester	79.5%	72.0%	85.7%	79.0%	80.0%	100.0	46.3
% Utilization assembler	83.5%	77.8%	89.0%	83.0%	84.0%	100.0	30.3
% Utilization final inspector	81.7%	73.4%	88.5%	81.1%	82.3%	100.0	52.8
Units packed - Throughput	821.4	752.0	878.0	816.0	826.8	100.0	6.8
% Util. packers	31.4%	28.5%	33.6%	31.1%	31.6%	100.0	48.8
W.I.P.	11.6	7.4	24.1	11.0	12.1	100.0	21.3

Table 4.2 shows the simulation responses or output variables of interest for the "Warm-up Period Validation" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. It was concluded from this table that no additional replicates were required.

Table 4.2 Validation of the warm-up period – "Warm-up Period Validation" model.

Testing Area	Warm-up Period Validation						
	Average $\mu 2$	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	10.5%	9.7%	11.1%	10.5%	10.6%	100.0	33.1
% Up-time Test. Mach. - Tester #1	90.1%	82.6%	97.7%	89.5%	90.7%	100.0	44.5
% Up-time Test. Mach. - Tester #2	89.9%	81.2%	98.6%	89.3%	90.5%	100.0	45.0
% Util. Test. Pers. - Tester#1	18.4%	17.1%	19.5%	18.3%	18.5%	100.0	30.8
% Util. Test. Pers. - Tester#2	18.4%	16.9%	19.7%	18.3%	18.5%	100.0	33.2
% Up-time Draw Out tester	79.9%	71.8%	85.5%	79.3%	80.4%	100.0	46.2
% Utilization assembler	83.8%	76.9%	89.1%	83.3%	84.2%	100.0	32.0
% Utilization final inspector	82.2%	73.8%	88.4%	81.6%	82.7%	100.0	52.6
Units packed - Throughput	826.2	747.0	884.0	820.8	831.6	100.0	6.9
% Util. packers	31.5%	28.5%	34.2%	31.3%	31.7%	100.0	46.5
W.I.P.	11.5	7.9	27.3	10.9	12.0	100.0	21.8

Table 4.3 shows the results from the hypothesis tests for the validation of the warm-up period. It shows for each performance measure of interest the mean difference between the "Actual System" model and the "Warm-up Period Validation" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded from this table that there was not enough evidence to have concerns about the warm-up period used during the validation process of the "Actual System" model. Thus, concluding that there were no reasons to doubt about the previous verification and validation processes of the model. This conclusion opened the door for the continuation of the simulation output analysis.

Table 4.3 Validation of the warm-up period – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	**Conf. Int. Min.**	**Conf. Int. Max.**	**Comments**
% Up-time Dielectric machine	0.01%	-0.07%	0.09%	Fail to Reject Ho
% Up-time Test. Mach. - Tester #1	-0.11%	-0.90%	0.68%	Fail to Reject Ho
% Up-time Test. Mach. - Tester #2	0.35%	-0.47%	1.16%	Fail to Reject Ho
% Util. Test. Pers. – Tester#1	-0.04%	-0.17%	0.10%	Fail to Reject Ho
% Util. Test. Pers. - Tester#2	-0.01%	-0.15%	0.13%	Fail to Reject Ho
% Up-time Draw Out tester	-0.37%	-1.09%	0.35%	Fail to Reject Ho
% Utilization assembler	-0.25%	-0.83%	0.34%	Fail to Reject Ho
% Utilization final inspector	-0.41%	-1.18%	0.35%	Fail to Reject Ho
Units packed – Throughput	-4.8	-11.91	2.31	Fail to Reject Ho
% Utilization packers	-0.16%	-0.45%	0.12%	Fail to Reject Ho
W.I.P.	0.0893	-0.5277	0.7063	Fail to Reject Ho

4.2.2. Sensitivity Analysis for Critical Input Parameters (Factors)

Sensitivity analyses were done with the objective of finding out input parameters of the model that had no major impact on the performance measures of interest or to the objectives of the project, and to define the experimental levels for the critical parameters. In summary, four sensitivity analyses were done. Results and conclusions are explained ahead, starting in the next page.

4.2.2.1.“Actual System” vs. “3000 Units Through Test”

The assumption of normality for the simulation output variable “Throughput” for the “Actual System” was previously validated. Please refer to section 4.2.1 and Figure 4.1.

The assumption of normality for the simulation output variable "Throughput" for the "3000 Units Through Test" model was validated by the normality test (refer to Figure 4.3). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

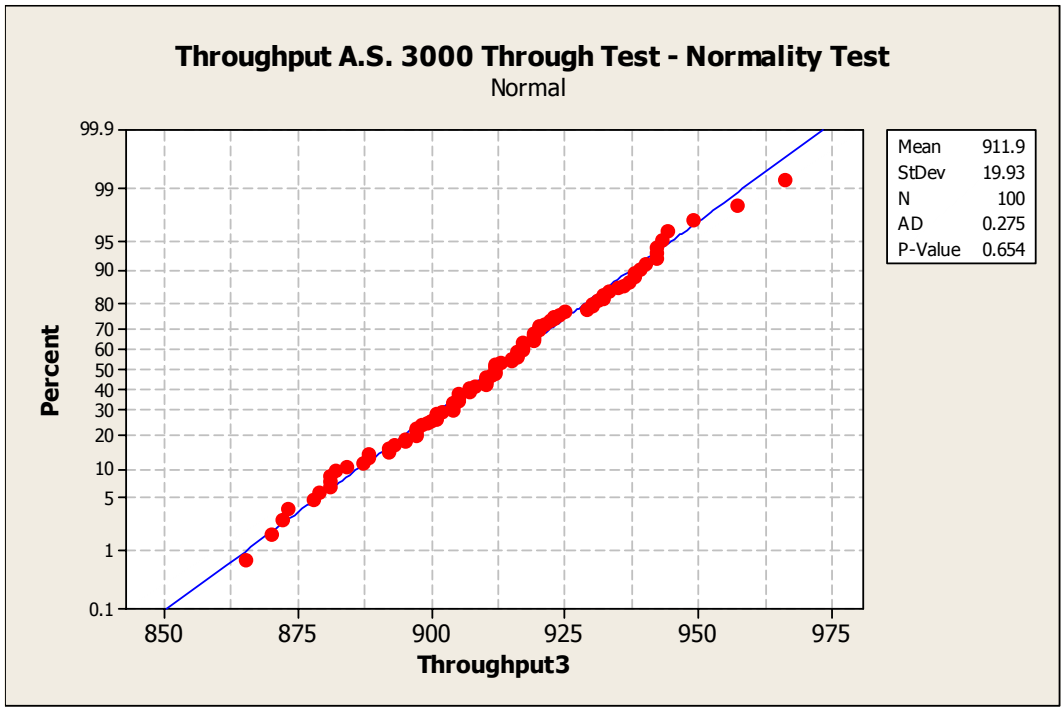

Figure 4.3 Normality test – "3000 Units Through Test" model

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

No additional replicates were required for the "Actual System" model. Please refer to section 4.2.1 and Table 4.1.

Table 4.4 shows the simulation responses or output variables of interest for "3000 Units Through Test" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum

number of runs required for achieving the desired confidence interval accuracy for the average value.

Table 4.4 "3000 Units Through Test" – required number of runs.

Testing Area	3000 Units Through Test						
	Average $\mu 2$	Min.	Max.	Conf. Int. Min.	Conf. Int. Max.	# Runs Mod	# Runs Est.
% Up-time Dielectric machine	26.5%	25.3%	27.8%	26.4%	26.6%	100.0	13.9
% Up-time Test. Mach. - Tester #1	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Up-time Test. Mach. - Tester #2	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Util. Test. Pers. - Tester#1	28.0%	27.2%	28.9%	28.0%	28.1%	100.0	5.5
% Util. Test. Pers. - Tester#2	28.0%	26.7%	29.0%	28.0%	28.1%	100.0	6.8
% Up-time Draw Out tester	88.7%	82.7%	93.9%	88.3%	89.0%	100.0	17.1
% Utilization assembler	91.5%	87.0%	96.2%	91.1%	91.8%	100.0	13.7
% Utilization final inspector	91.6%	85.4%	96.8%	91.2%	92.0%	100.0	17.6
Units packed - Throughput	911.9	865.0	966.0	907.8	915.9	100.0	3.1
% Util. packers	34.8%	32.8%	37.2%	34.6%	35.0%	100.0	22.7
W.I.P.	986.5	899.5	1057.9	980.0	993.1	100.0	0.4

The following was concluded from this table:

- The number of dielectric machines was not considered as a critical factor for the achievement of the goals of this project. This was concluded after observing the low percentage of up-time at 3000 units already.
- Note that the up-time for testing machines #1 and #2 showed no variability, thus the number of runs was estimated as 0. The main reason for this was that at 3000 units, the testing machines became the bottleneck as they achieved 100% up-time for all replicates. This was also the main reason for the high number of units in work-in-process. The conclusion from these observations

was that the testing machines were critical factors to be included in further experimental designs.

- Note that the testing personnel utilization (#1 and #2) is at 28% while the utilization for the assembly and final inspection personnel is at 91.5% and 91.6%, respectively. This suggests a labor balancing (work load) problem related to the execution of the task named Inspection I (refer to Figure 1.1). It was decided to add the balancing issue as a critical factor for further consideration in experimental designs. This factor will consider the location of Inspection I at the testing area vs. draw out/final inspection area.

The need for adding testing machines was also identified during the previous project. At that time, the possibility for adding machines raised a labor related concern if the total number of machines required was an odd number, and multiple machines per employee were suggested due to the possibility of having employees with the same classification not serving the same number of machines. The reason for the concern was that testing personnel used to work independently; they were not supporting each other. At that time, management decided that testing personnel must support each other and no longer were they working at only one machine.

Returning to this project, the above explanation was the reason for modifying the model "3000 Units Through Test" to reflect the change in the company policy, in other words, any entity at any of the testing area processes could seize available units from the resources named testing personnel. This new or updated model was used for the sensitivity analysis at the testing station (refer to section 3.2.2.2 for the methodology and section 4.2.2.2 for the output analysis).

Table 4.5 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "Actual System" model and the "3000 Units Through Test" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that the model was sensitive to the inter-arrival rate since H_0 was rejected for all variables. These results suggested the inclusion of this parameter in further experimental designs.

Table 4.5 "Actual System" vs. "3000 Units Through Test" – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	Conf. Int. Min.	Conf. Int. Max.	Comments
% Up-time Dielectric machine	-15.90%	-16.02%	-15.78%	Reject Ho
% Up-time Test. Mach. - Tester #1	-10.00%	-10.60%	-9.40%	Reject Ho
% Up-time Test. Mach. - Tester #2	-9.78%	-10.39%	-9.17%	Reject Ho
% Util. Test. Pers. – Tester #1	-9.69%	-9.81%	-9.57%	Reject Ho
% Util. Test. Pers. – Tester #2	-9.65%	-9.79%	-9.51%	Reject Ho
% Up-time Draw Out tester	-9.18%	-9.88%	-8.48%	Reject Ho
% Utilization assembler	-7.94%	-8.54%	-7.34%	Reject Ho
% Utilization final inspector	-9.88%	-10.65%	-9.11%	Reject Ho
Units packed - Throughput	-90.4	-97.53	-83.27	Reject Ho
% Util. packers	-3.45%	-3.74%	-3.16%	Reject Ho
W.I.P.	-975	-981.48	-968.52	Reject Ho

4.2.2.2.Testing Area

4.2.2.2.1. The following is the analysis done to determine if the number of testing machines was a critical factor:

The assumption of normality for the simulation output variable "Throughput" for the "New System 3000 Units Through Test" model was validated by the normality test (refer to Figure 4.4). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

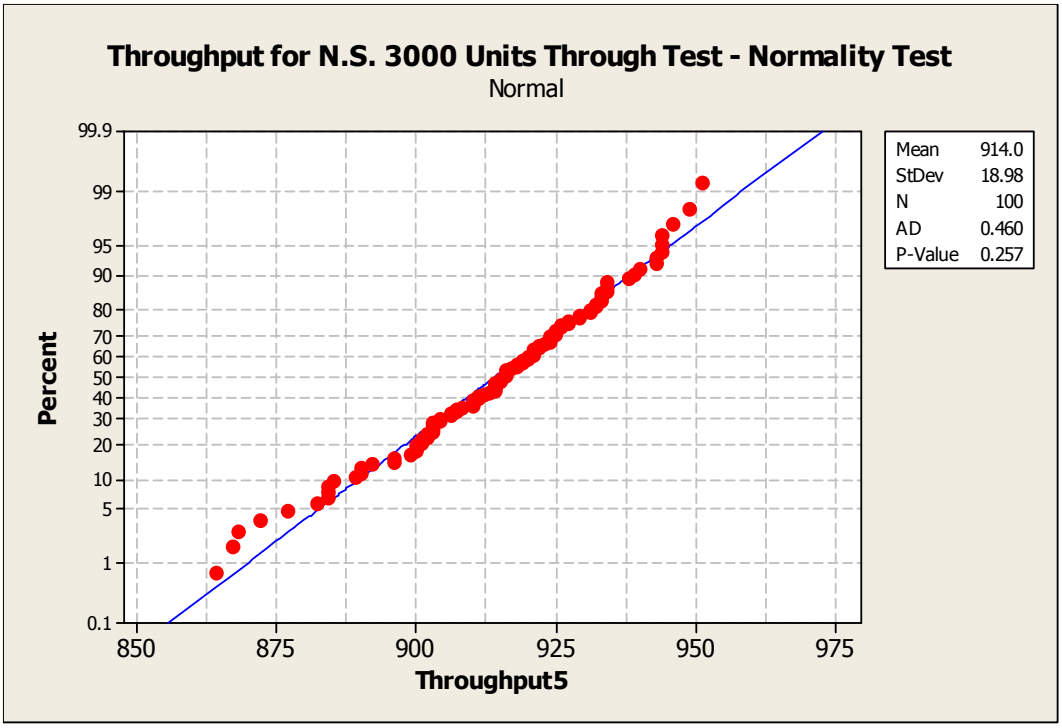

Figure 4.4 Normality test – "New System 3000 Units Through Test" model.

The assumption of normality for the simulation output variable "Throughput" for the "6 Testing Machine" model was validated by the normality test (refer to Figure 4.5). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

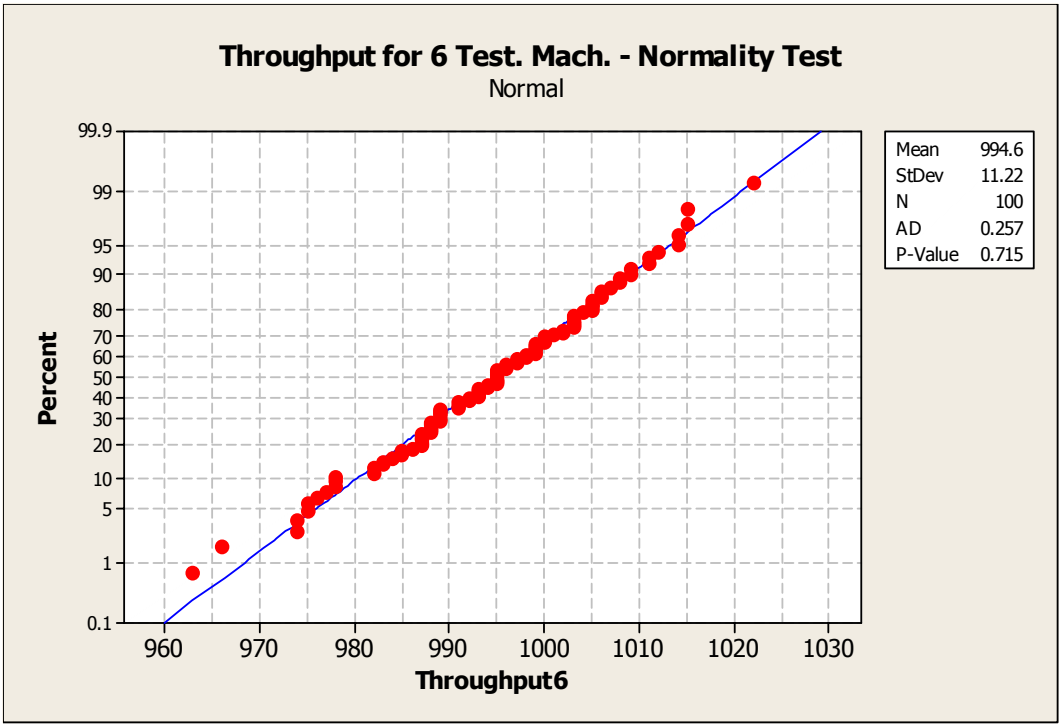

Figure 4.5 Normality test – "6 Testing Machines" model.

The conclusions from Table 4.6 were the same as of Table 4.4. This was expected because the only change between model "3000 Units Through Test" and model "New System 3000 Units Through Test" was the new company policy related to the testing area. Note that the up-time for the testing machines showed no variability, thus the number of runs was estimated as 0. The main reason for this was that at 3000 units, the testing machines became the bottleneck as they achieved 100% up-time for all replicates. This was also the explanation for the high number of units in work-in-process. In summary, the conclusions were the following: the number of dielectric test machines was not considered a critical factor, the number of testing machines was considered a critical factor, and the labor balancing issue between the testing area and draw out/final inspection area was also considered a critical factor. It was also concluded that no more replications were needed.

Table 4.6 "New System 3000 Units Through Test" – required number of runs.

Testing Area	3000 Units Through Test						
	Average μ1	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	27.1%	25.9%	28.6%	27.0%	27.2%	100.0	14.1
% Up-time Test. Mach. - Tester #1	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Util. Test. Pers. – Tester #1	28.1%	27.5%	28.8%	28.1%	28.2%	100.0	4.4
% Up-time Draw Out tester	89.0%	84.7%	93.0%	88.7%	89.4%	100.0	15.6
% Utilization assembler	91.8%	87.3%	95.7%	91.5%	92.1%	100.0	13.1
% Utilization final inspector	92.0%	87.9%	96.1%	91.6%	92.4%	100.0	16.3
Units packed – Throughput	914.0	864.0	951.0	910.2	917.8	100.0	2.8
% Utilization packers	34.9%	32.8%	36.6%	34.7%	35.1%	100.0	21.0
W.I.P.	981.8	892.0	1052.3	975.2	988.4	100.0	0.5

Table 4.7 shows the simulation responses or output variables of interest for "6 Testing Machines" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The following was concluded from this table:

- No additional replication was required for the output analysis.
- At this stage, the number of testing personnel was not identified as a critical factor. Please note the low utilization of the personnel at 3000 units already.
- The draw out/final inspection area became the bottleneck as the testing area increased the number of units processed due to the increment in testing machines. Note that the utilization for the assembly and final inspection personnel showed no variability, thus the number of runs was

estimated as 0. The main reason for this was that 100% utilization was achieved for all replicates. This was also the explanation for the high number of units in work-in-process. The number of assembly and final inspection personnel were identified as critical factors.

Table 4.7 "6 Testing Machine" – required number of runs.

	6 Testing Machines						
Testing Area	**Average $\mu 2$**	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	29.8%	28.1%	31.8%	29.6%	29.9%	100.0	26.4
% Up-time Test. Mach. - Tester #1	75.4%	72.1%	78.8%	75.1%	75.6%	100.0	14.6
% Util. Test. Pers. – Tester #1	45.9%	44.0%	48.0%	45.7%	46.1%	100.0	12.9
% Up-time Draw Out tester	96.7%	95.8%	97.7%	96.6%	96.8%	100.0	0.7
% Utilization assembler	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Utilization final inspector	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
Units packed - Throughput	994.6	963.0	1022.0	992.3	996.8	100.0	0.8
% Utilization packers	38.0%	36.8%	39.0%	37.9%	38.1%	100.0	6.9
W.I.P.	664.3	579.8	727.6	658.8	669.8	100.0	0.7

Table 4.8 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "New System 3000 Units Through Test" model and the "6 Testing Machines" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that the model was sensitive to the number of testing machines since H_0 was rejected for all variables. These results suggested the inclusion of this parameter in further experimental designs

Table 4.8 "New System 3000 Units Through Test" vs. "6 Testing Machines" – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	Conf. Int. Min.	Conf. Int. Max.	Comments
% Up-time Dielectric machine	-2.63%	-2.70%	-2.56%	Reject Ho
% Up-time Test. Mach. - Tester #1	24.60%	24.32%	24.89%	Reject Ho
% Util. Test. Pers. – Tester #1	-17.80%	-17.92%	-17.68%	Reject Ho
% Up-time Draw Out tester	-7.64%	-8.00%	-7.28%	Reject Ho
% Utilization assembler	-8.19%	-8.52%	-7.86%	Reject Ho
% Utilization final inspector	-7.99%	-8.36%	-7.62%	Reject Ho
Units packed - Throughput	-80.5	-84.88	-76.12	Reject Ho
% Utilization packers	-3.06%	-3.24%	-2.88%	Reject Ho
W.I.P.	317	312.97	321.03	Reject Ho

4.2.2.2.2. The following is the analysis done to determine if the yield was a critical factor:

The assumption of normality for the simulation output variable "Throughput" for the "New System 3000 Units Through Test" was previously validated. Please refer to Figure 4.4.

The assumption of normality for the simulation output variable "Throughput" for the "90% Yield" model was validated by the normality test (refer to Figure 4.6). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

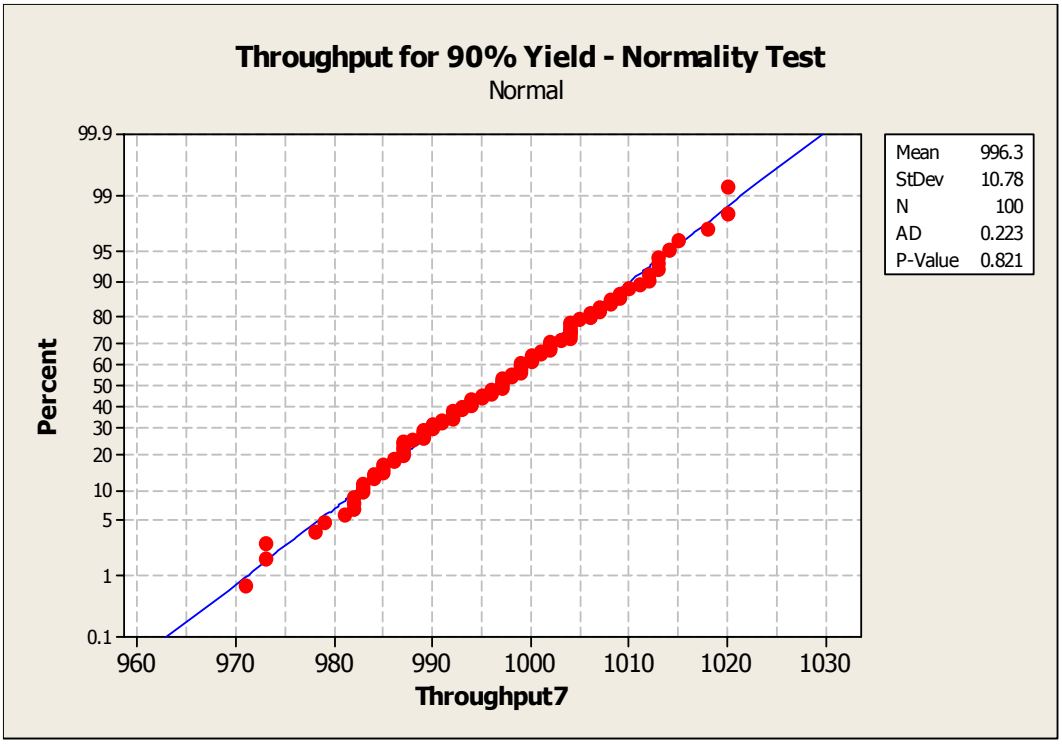

Figure 4.6 Normality test – "90% Yield" model.

As for the number of required replicates for the model "New System 3000 Units Through Test" refer to Table 4.6. It was already concluded that no additional replications were needed.

Table 4.9 shows the simulation responses or output variables of interest for "90% Yield" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The following was concluded:

- No additional replicates were required.
- The results confirmed the criticality of the number of testing machines. Note that the up-time for the testing machines showed no variability, thus the number of runs was estimated as 0. The main reason for this was that the testing machines became the bottleneck as they achieved 100% up-

time for all replicates. This contributed to the high number of units in work-in-process.

- The results confirmed the criticality for the number of final inspectors and assemblers as they became the reason for the bottleneck at the draw out/final inspection area. Note that the utilization for the assembly and final inspection personnel showed almost no variability, thus the estimate for the number of runs was rounded to 0. The main reason for this was that 100% utilization was achieved for almost all replicates; in fact the minimum utilization for a replicate was 99.8% for both performance measures, which can be rounded to 100%. This contributed also to the high number of units in work-in-process.

Table 4.9 "90% Yield" – required number of runs.

	90% Yield						
Testing Area	**Average μ2**	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	27.4%	26.0%	28.8%	27.3%	27.5%	100.0	14.2
% Up-time Test. Mach. - Tester #1	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Util. Test. Pers. – Tester #1	29.2%	28.5%	30.0%	29.1%	29.2%	100.0	3.9
% Up-time Draw Out tester	96.7%	95.6%	97.6%	96.7%	96.8%	100.0	0.6
% Utilization assembler	100.0%	99.8%	100.0%	100.0%	100.0%	100.0	0.0
% Utilization final inspector	100.0%	99.8%	100.0%	100.0%	100.0%	100.0	0.0
Units packed – Throughput	996.3	971.0	1020.0	994.2	998.5	100.0	0.8
% Utilization packers	38.0%	37.0%	39.0%	37.9%	38.1%	100.0	6.0
W.I.P.	1067.9	977.1	1145.6	1061.2	1074.6	100.0	0.4

Table 4.10 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "New System 3000 Units Through Test" model and the "90% Yield" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that Yield was a critical input parameter since H_0 was rejected for almost all of the performance measures. Note that no significant difference was observed in the case of

the up-time for the testing machines; but this result was mostly driven by the fact that the up-time was 100% for all replicates under both models, showing that additional testing machines were required. These results suggested the inclusion of this parameter in further experimental designs.

Table 4.10 "New System 3000 Units Through Test" vs. "90% Yield" – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	**Conf. Int. Min.**	**Conf. Int. Max.**	**Comments**
% Up-time Dielectric machine	-0.24%	-0.27%	-0.21%	Reject Ho
% Up-time Test. Mach. - Tester #1	0.00%	0.00%	0.00%	Fail to Reject Ho
% Util. Test. Pers. – Tester #1	-1.05%	-1.09%	-1.01%	Reject Ho
% Up-time Draw Out tester	-7.69%	-8.04%	-7.34%	Reject Ho
% Utilization assembler	-8.19%	-8.52%	-7.86%	Reject Ho
% Utilization final inspector	-7.98%	-8.35%	-7.61%	Reject Ho
Units packed - Throughput	-82.3	-86.7	-77.9	Reject Ho
% Utilization packers	-3.14%	-3.32%	-2.96%	Reject Ho
W.I.P.	-86.1	-88.11	-84.09	Reject Ho

4.2.2.2.3. The following is the analysis done to determine if the testing process time was a critical factor:

The assumption of normality for the simulation output variable "Throughput" for the "New System 3000 Units Through Test" was previously validated. Please refer to Figure 4.4.

The assumption of normality for the simulation output variable "Throughput" for the "30% Reduction Testing Process Time" model was validated by the normality test (refer to Figure 4.7). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

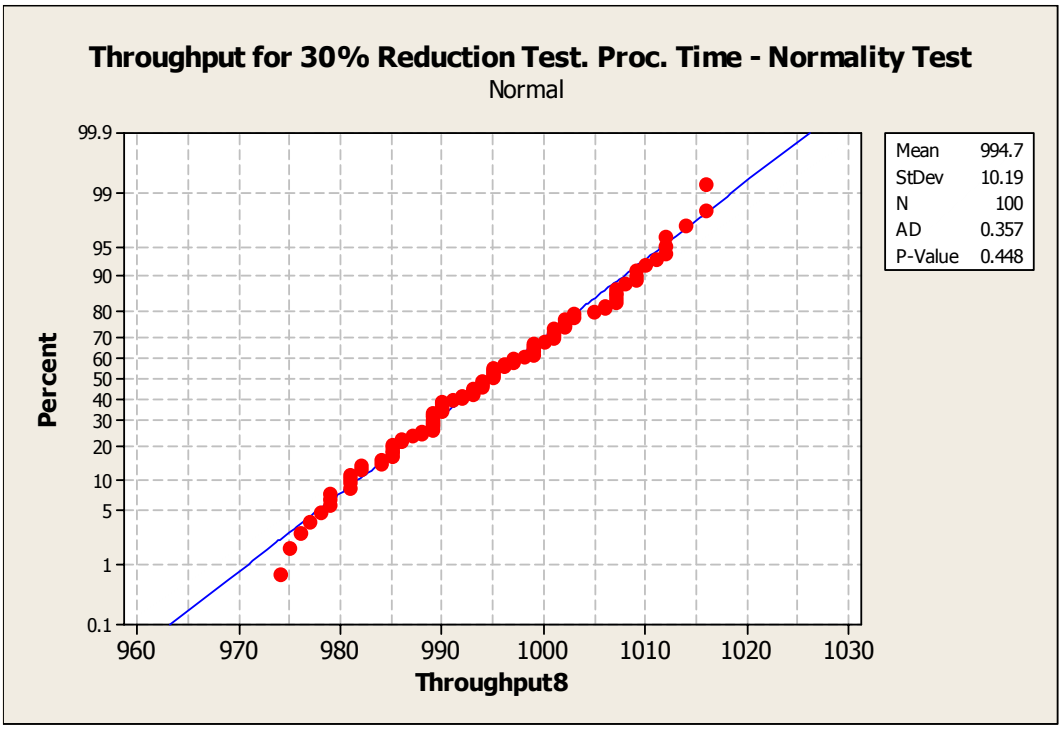

Figure 4.7 Normality test – "30% Reduction Testing Process Time" model.

As for the number of required replicates for the model "New System 3000 Units Through Test", refer to Table 4.6. It was already concluded that no additional replications were needed.

Table 4.11 shows the simulation responses or output variables of interest for "30% Reduction Testing Process Time" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The conclusions were the following:

- No additional replicates were required.

- The results confirmed the criticality of the number of testing machines. Note that the up-time for the testing machines showed no variability, thus the number of runs was estimated as 0. The main reason for this was that the testing machines became the bottleneck as they achieved 100% up-time for all replicates. This contributed to the high number of units in work-in-process.
- The number of final inspectors and assemblers became the reason for the bottleneck at the draw out/final inspection area. Note that the utilization for the assembly and final inspection personnel showed no variability, thus the number of runs was estimated as 0. The main reason for this was that 100% utilization was achieved for all replicates. This contributed also to the high number of units in work-in-process.

Table 4.11 "30% Reduction Testing Process Time" – required number of runs.

	30% Reduction Testing Processing Time						
Testing Area	**Average μ2**	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	27.7%	26.3%	29.1%	27.6%	27.8%	100.0	15.5
% Up-time Test. Mach. - Tester #1	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Util. Test. Pers. – Tester #1	33.4%	32.6%	34.2%	33.3%	33.5%	100.0	3.0
% Up-time Draw Out tester	96.8%	95.9%	97.6%	96.7%	96.9%	100.0	0.7
% Utilization assembler	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Utilization final inspector	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
Units packed – Throughput	994.7	974.0	1016.0	992.7	996.8	100.0	0.7
% Utilization packers	38.0%	36.6%	38.9%	37.9%	38.1%	100.0	6.0
W.I.P.	851.7	740.7	922.7	844.4	859.0	100.0	0.7

Table 4.12 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "New System 3000 Units Through Test" model and the "30% Reduction Testing Process Time" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that the model was sensitive to the testing process time since H_0 was rejected for almost all of the performance measures. Note that no significant difference was observed in the case of the up-time for the testing machines; but this result was mostly driven by the fact that the up-time was 100% for all replicates under both models, showing that additional testing machines were required. These results suggested the inclusion of this parameter in further experimental designs.

Table 4.12 "New System 3000 Units Through Test" vs. "30% Red. Test Proc. Time" – Hypothesis test

	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
Testing Area	**Mean Diff.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**Comments**
% Up-time Dielectric machine	-0.58%	-0.61%	-0.54%	Reject Ho
% Up-time Test. Mach. - Tester #1	0.00%	0.00%	0.00%	Fail to Reject Ho
% Util. Test. Pers. – Tester #1	-5.28%	-5.32%	-5.24%	Reject Ho
% Up-time Draw Out tester	-7.75%	-8.11%	-7.40%	Reject Ho
% Utilization assembler	-8.19%	-8.52%	-7.86%	Reject Ho
% Utilization final inspector	-7.99%	-8.36%	-7.62%	Reject Ho
Units packed - Throughput	-80.7	-84.82	-76.58	Reject Ho
% Utilization packers	-3.08%	-3.25%	-2.91%	Reject Ho
W.I.P.	130	127.04	132.96	Reject Ho

4.2.2.3.Final Inspection Area

4.2.2.3.1. The following is the analysis done to corroborate if the number of assemblers was a critical factor:

The assumption of normality for the simulation output variable "Throughput" for the "New System 3000 Units Through Draw Out" model was validated by the normality test (refer to Figure 4.8). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

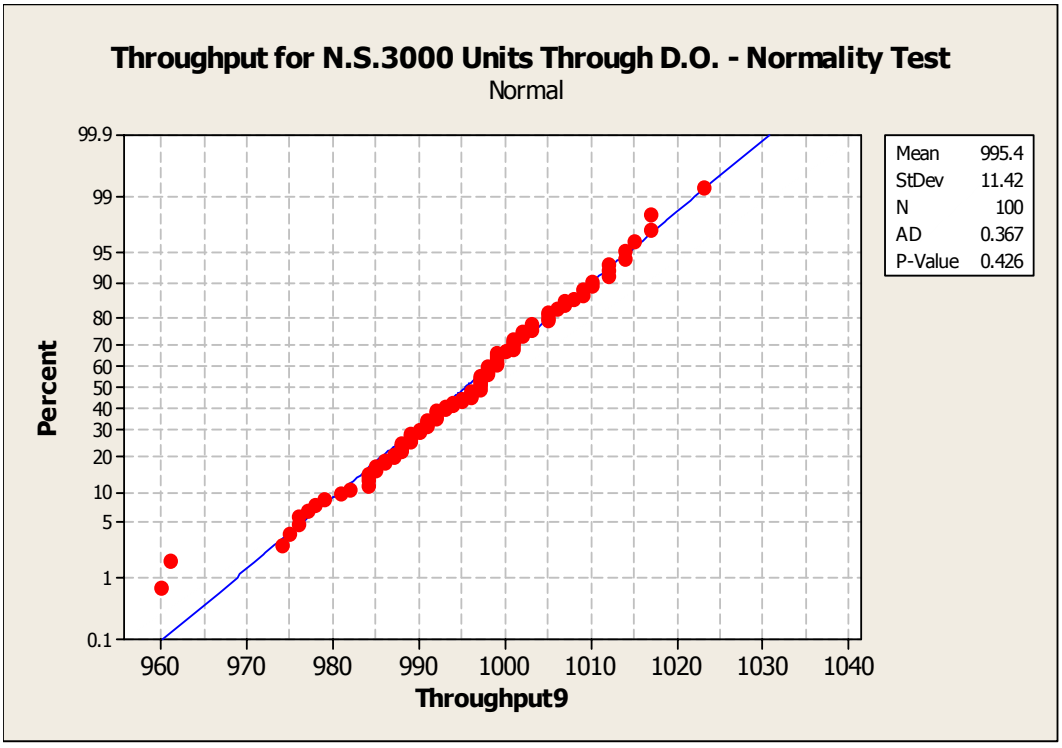

Figure 4.8 Normality test – "New System at 3000 Units Through Draw Out" model.

The assumption of normality for the simulation output variable "Throughput" for the "4 Assemblers" model was validated by the normality test (refer to Figure 4.9). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

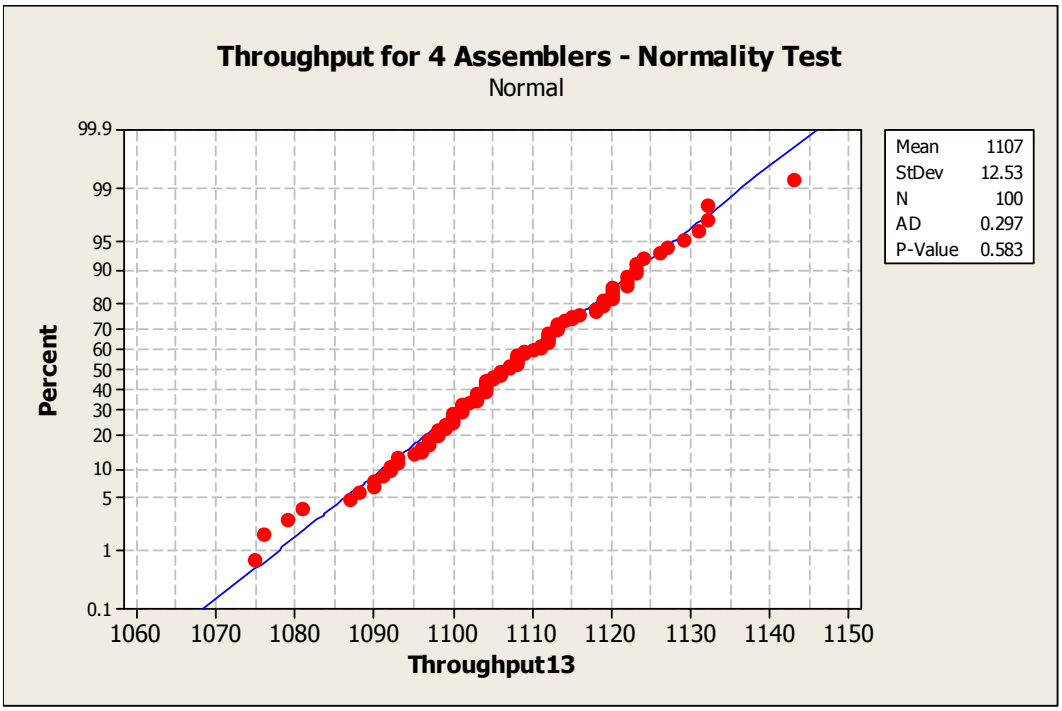

Figure 4.9 Normality test – "4 Assemblers" model

Table 4.13 shows the simulation responses or output variables of interest for "New System 3000 Units Through Test" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The conclusions were the following:

- No additional replicates were required.
- The results confirmed the criticality of the number of final inspectors and assemblers as they became the reason for the bottleneck at the draw out/final inspection area. Note that the utilization for the assembly and

final inspection personnel showed no variability, thus the number of runs was estimated as 0. The main reason for this was that 100% utilization was achieved for all replicates. This was the main reason for the high number of units in work-in-process.

Table 4.13 "New System 3000 Units Through Draw Out" – required number of runs.

Testing Area	3000 Units Through Draw Out						
	Average μ1	Min.	Max.	Conf. Int. Min.	Conf. Int. Max.	# Runs Mod	# Runs Est.
% Up-time Dielectric machine	32.2%	30.2%	34.7%	32.0%	32.3%	100.0	25.0
% Up-time Test. Mach. - Tester #1	58.3%	55.8%	61.3%	58.1%	58.5%	100.0	13.2
% Util. Test. Pers. – Tester #1	50.8%	48.5%	53.5%	50.7%	51.0%	100.0	12.1
% Up-time Draw Out tester	96.8%	95.6%	97.8%	96.7%	96.8%	100.0	0.7
% Utilization assembler	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Utilization final inspector	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
Units packed - Throughput	995.4	960.0	1023.0	993.1	997.7	100.0	0.9
% Utilization packers	38.0%	36.5%	39.2%	37.9%	38.1%	100.0	6.7
W.I.P.	1050.4	953.3	1143.6	1043.2	1057.6	100.0	0.5

Table 4.14 shows the simulation responses or output variables of interest for "4 Assemblers" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The conclusions were the following:

- No additional replicates were required.
- Note that the up-time for the draw out machine showed no variability, thus the number of runs was estimated as 0. The main reason for this was that the draw out machine became the bottleneck as it achieved 100% up-time for all replicates. This was the main reason for the high number of units in

work-in-process. This suggested the inclusion of the number of draw out machine into the critical factor list.

Table 4.14 "4 Assemblers" – required number of runs.

Testing Area	4 Assembly Personnel						
	Average $\mu 2$	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	32.1%	30.0%	34.3%	32.0%	32.3%	100.0	26.3
% Up-time Test. Mach. - Tester #1	58.3%	55.7%	61.2%	58.1%	58.5%	100.0	13.8
% Util. Test.Pers. - Tester#1	50.8%	48.5%	53.3%	50.7%	51.0%	100.0	12.1
% Up-time Draw Out tester	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Utilization assembler	66.5%	65.0%	68.8%	66.4%	66.7%	100.0	5.0
% Utilization final inspector	68.4%	66.3%	72.0%	68.1%	68.6%	100.0	9.6
Units packed - Throughput	1107.1	1075.0	1143.0	1104.6	1109.6	100.0	0.8
% Utilization packers	42.3%	40.9%	43.5%	42.2%	42.4%	100.0	6.3
W.I.P.	957.3	857.2	1032.5	950.0	964.6	100.0	0.6

Table 4.15 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "New System 3000 Units Through Draw Out" model and the "4 Assemblers" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that the model was sensitive to the number of assembly personnel since H_0 was rejected at almost all of the performance measures. No significant difference was observed in the case of the performance measures related to the testing area, but this was expected since the products go first to the testing area and then to the draw out/final inspection area. These results suggested the inclusion of this parameter in further experimental designs.

Table 4.15 "New System 3000 Units Through Draw Out" vs. "4 Assemblers" – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	**Conf. Int. Min.**	**Conf. Int. Max.**	**Comments**
% Up-time Dielectric machine	0.03%	-0.03%	0.10%	Fail to Reject Ho
% Up-time Test. Mach. - Tester #1	-0.03%	-0.12%	0.06%	Fail to Reject Ho
% Util. Test.Pers. - Tester#1	0.00%	-0.03%	0.02%	Fail to Reject Ho
% Up-time Draw Out tester	-3.25%	-3.33%	-3.17%	Reject Ho
% Utilization assembler	33.50%	33.35%	33.65%	Reject Ho
% Utilization final inspector	31.60%	31.39%	31.81%	Reject Ho
Units packed - Throughput	-112	-115.24	-108.76	Reject Ho
% Utilization packers	-4.26%	-4.40%	-4.12%	Reject Ho
W.I.P.	93.2	90.73	95.67	Reject Ho

4.2.2.3.2. The following is the analysis done to corroborate if the number of final inspectors was a critical factor:

The assumption of normality for the simulation output variable "Throughput" for the "New System 3000 Units Through Draw Out" was previously validated. Please refer to Figure 4.8.

The assumption of normality for the simulation output variable "Throughput" for the "4 final inspectors" model was validated by the normality test (refer to Figure 4.10). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

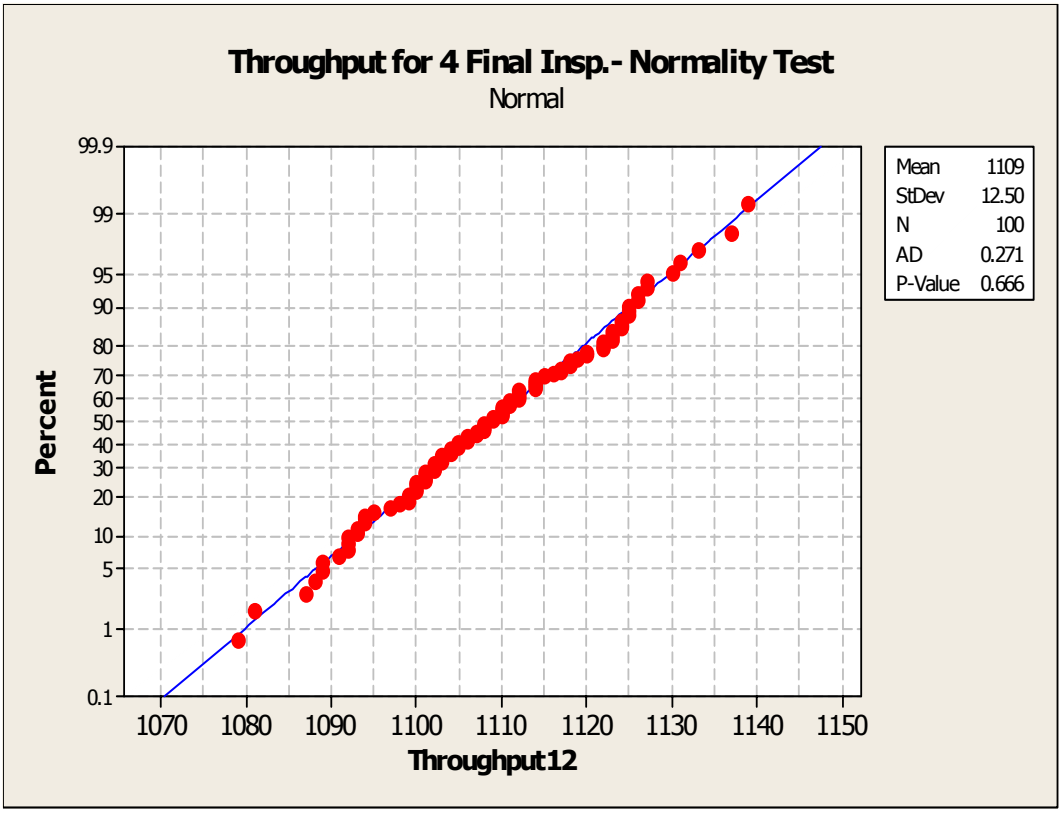

Figure 4.10 Normality test – "4 Final Inspectors" model

As for the number of required replicates for the model "New System 3000 Units Through Draw Out", refer to Table 4.13. It was already concluded that no additional replications were needed.

Table 4.16 shows the simulation responses or output variables of interest for "4 Final Inspectors" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The conclusions were the following:

- No additional replicates were required.
- Note that the up-time for the draw out machine showed no variability, thus the number of runs was estimated as 0. The main reason for this was that the draw out machine became the bottleneck as it achieved 100% up-time

for all replicates. This was the main reason for the high number of units in work-in-process. This suggested the number of draw out machines as a critical factor.

Table 4.16 "4 Final Inspectors" – required number of runs.

Testing Area	4 Final Inspectors						
	Average $\mu 2$	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	32.1%	30.0%	34.2%	31.9%	32.3%	100.0	26.7
% Up-time Test. Mach. - Tester #1	58.3%	55.3%	61.3%	58.1%	58.5%	100.0	15.3
% Util. Test. Pers. – Tester #1	50.8%	48.6%	53.5%	50.7%	51.0%	100.0	11.8
% Up-time Draw Out tester	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Utilization assembler	78.6%	77.3%	80.3%	78.5%	78.7%	100.0	2.1
% Utilization final inspector	64.0%	62.3%	66.2%	63.8%	64.2%	100.0	6.3
Units packed - Throughput	1108.9	1079.0	1139.0	1106.4	1111.4	100.0	0.8
% Utilization packers	42.3%	40.8%	43.6%	42.2%	42.4%	100.0	7.5
W.I.P.	956.8	863.0	1032.9	949.6	964.0	100.0	0.6

Table 4.17 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "New System 3000 Units Through Draw Out" model and the "4 Final Inspectors" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that the model was sensitive to the number of final inspectors since H_0 was rejected for almost all of the performance measures. No significant difference was observed in the case of the performance measures related to the testing area (except for the up-time of the dielectric test), but this was expected since the products go first to the testing area and then to the draw out/final inspection area. In the case of the dielectric machine, the difference was not related to the number of final inspectors, but rather to the need for controlling the random numbers. These results suggested the inclusion of this parameter (number of final inspectors) in further experimental designs.

Table 4.17 "New System 3000 Units Through Draw Out" vs. "4 final inspectors" – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	**Conf. Int. Min.**	**Conf. Int. Max.**	**Comments**
% Up-time Dielectric machine	0.07%	0.00%	0.13%	Reject Ho
% Up-time Test. Mach. - Tester #1	-0.02%	-0.11%	0.08%	Fail to Reject Ho
% Util. Test. Pers. – Tester #1	-4.61E-05	-0.03%	0.02%	Fail to Reject Ho
% Up-time Draw Out tester	-3.25%	-3.33%	-3.17%	Reject Ho
% Utilization assembler	21.40%	21.29%	21.51%	Reject Ho
% Utilization final inspector	36.00%	35.84%	36.16%	Reject Ho
Units packed - Throughput	-114	-117.49	-110.51	Reject Ho
% Utilization packers	-4.33%	-4.48%	-4.18%	Reject Ho
W.I.P.	93.7	91.07	96.33	Reject Ho

4.2.2.3.3. The following is the analysis done to corroborate if the number of draw out machines was a critical factor:

The assumption of normality for the simulation output variable "Throughput" for the "New System 3000 Units Through Draw Out" was previously validated. Please refer to Figure 4.8.

The assumption of normality for the simulation output variable "Throughput" for the "3 Draw Out Machines" model was validated by the normality test (refer to Figure 4.11). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

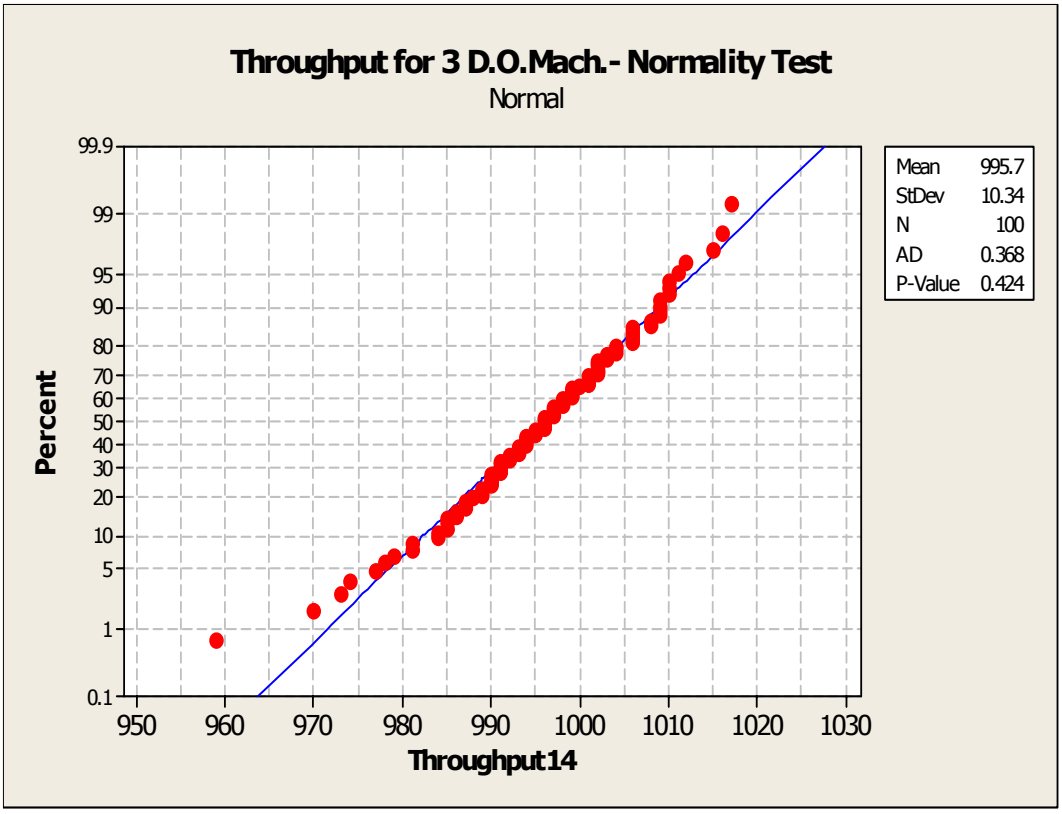

Figure 4.11 Normality test – "3 Draw Out Machines" model

As for the number of required replicates for the model "New System 3000 Units Through Draw Out", refer to Table 4.13. It was already concluded that no additional replications were needed.

Table 4.18 shows the simulation responses or output variables of interest for "3 Draw Out Machines" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The conclusions were the following:

- No additional replicates were required.
- The results confirmed the criticality of the number of final inspectors and assemblers as they became the reason for the bottleneck at the draw out/final inspection area. Note that the utilization for the assembly and

final inspection personnel showed no variability, thus the number of runs was estimated as 0. The main reason for this was that 100% utilization was achieved for all replicates. This was the main reason for the high number of units in work-in-process.

Table 4.18 "3 Draw Out Machines" – required number of runs.

Testing Area	3 Draw Out Machines						
	Average $\mu 2$	Min.	Max.	Conf. Int. Min.	Conf. Int. Max.	# Runs Mod	# Runs Est.
% Up-time Dielectric machine	32.2%	29.8%	34.1%	32.0%	32.3%	100.0	26.3
% Up-time Test. Mach. - Tester #1	58.3%	55.7%	61.1%	58.1%	58.5%	100.0	12.5
% Util. Test. Pers. – Tester #1	50.9%	48.6%	53.4%	50.7%	51.0%	100.0	12.4
% Up-time Draw Out tester	46.0%	45.2%	46.8%	46.0%	46.1%	100.0	1.5
% Utilization assembler	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
% Utilization final inspector	100.0%	100.0%	100.0%	100.0%	100.0%	100.0	0.0
Units packed - Throughput	995.7	959.0	1017.0	993.6	997.8	100.0	0.7
% Utilization packers	38.0%	36.6%	39.1%	37.9%	38.1%	100.0	6.2
W.I.P.	1050.6	952.1	1140.4	1043.2	1058.0	100.0	0.5

Table 4.19 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "New System 3000 Units Through Draw Out" model and the "3 Draw Out Machines" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that the model was sensitive to the number of draw out machines since H_0 was rejected for at least one performance measure ("% Up-time Draw Out Tester"). The following are the explanations for the number of draw out machines not showing as critical for the other performance measures:

- No significant difference was observed with the performance measures related to the draw out/final inspection and packing areas, except for the draw out machine up-time. This was driven mostly by the fact that the final inspectors as well as the assemblers were causing a bottleneck at this

area. As a result, the benefit of adding draw out machines was not observed throughout the other metrics. Note that the utilization for the assembly and final inspection personnel showed no variability, thus the number of runs was estimated as 0. The main reason for this was that 100% utilization was achieved for all replicates.

- No significant difference was observed with the performance measures related to the testing area, but this was expected since the number of draw out machines had no impact on the testing area.

Table 4.19 "New System 3000 Units Through Draw Out" vs. "3 Draw Out Machines" – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	Conf. Int. Min.	Conf. Int. Max.	Comments
% Up-time Dielectric machine	-8.19E-05	-0.07%	0.05%	Fail to Reject Ho
% Up-time Test. Mach. - Tester #1	-0.04%	-0.08%	0.01%	Fail to Reject Ho
% Util. Test. Pers. – Tester #1	-0.01%	-0.03%	0.01%	Fail to Reject Ho
% Up-time Draw Out tester	50.70%	50.63%	50.77%	Reject Ho
% Utilization assembler	0.00%	0.00%	0.00%	Fail to Reject Ho
% Utilization final inspector	0.00%	0.00%	0.00%	Fail to Reject Ho
Units packed - Throughput	-0.24	-2.18	1.7	Fail to Reject Ho
% Utilization packers	-0.02%	-0.09%	0.06%	Fail to Reject Ho
W.I.P.	-0.195	-2.055	1.665	Fail to Reject Ho

4.2.2.4.Packing Area

The following is the analysis done to determine if the number of packing personnel was a critical factor:

The assumption of normality for the simulation output variable "Throughput" for the "New System 3000 Units Through Packing" model was validated by the normality test (refer to Figure 4.12). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

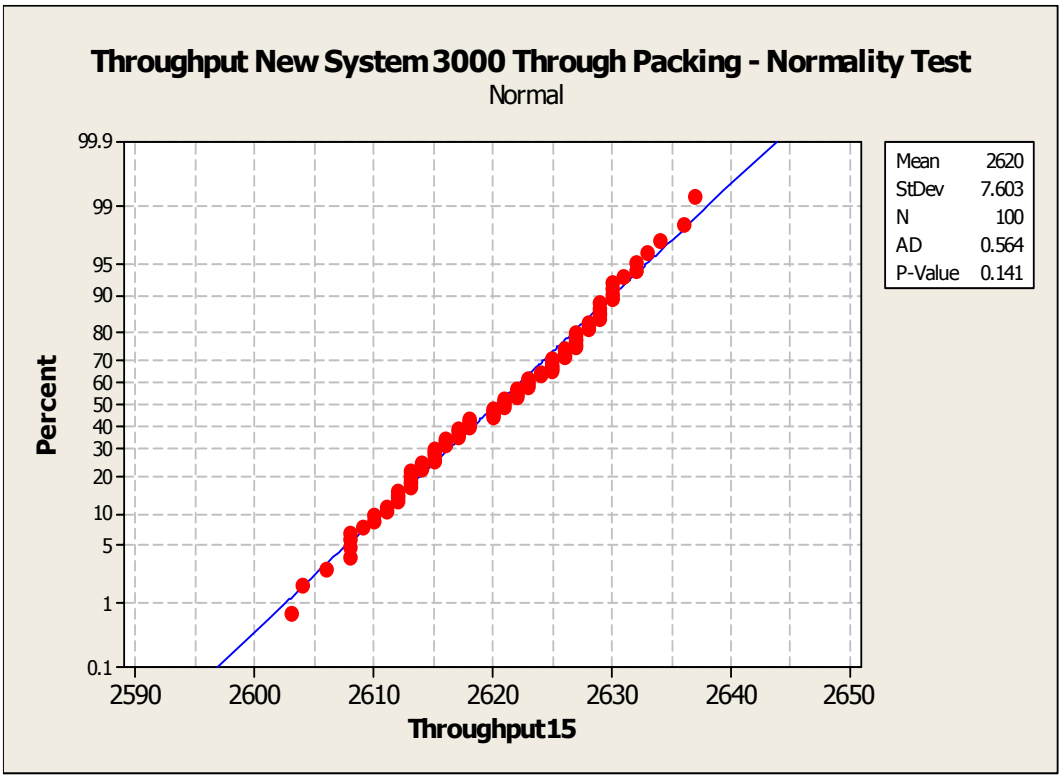

Figure 4.12 Normality test – "New System 3000 Units Through Packing" model.

The assumption of normality for the simulation output variable "Throughput" for the "3 Packers" model was validated by the normality test (refer to Figure 4.13). The hypothesis test was:

- H_0: Data is normally distributed.
- H_a: Data is not normally distributed.

Based on the *p-value* > 0.05, it was concluded that there was not enough evidence to reject H_0. Therefore, the assumption of the data being identically distributed prevailed, allowing the simulation output analysis to continue.

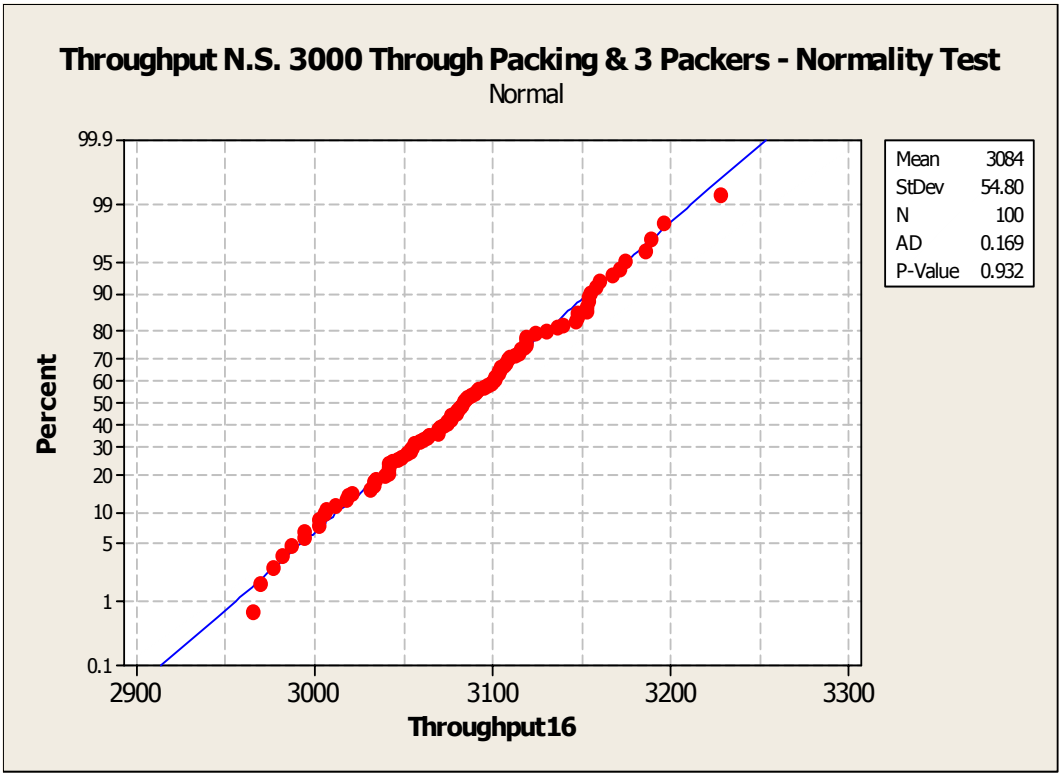

Figure 4.13 Normality test – "3 Packers" model.

Table 4.20 shows the simulation responses or output variables of interest for "New System 3000 Units Through Packing" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The conclusions were the following:

- No additional replicates were required.
- The results showed the number of packers as a critical factor as they became the bottleneck at the packing area. Note that the labor utilization for the packer reached 100% for almost all replicates, thus the estimate for the number of runs was rounded to 0. In fact, the minimum value was 99.7% which can be rounded to 100%. In average, 2620.3 units were packed, meaning that more packing personnel were required to achieve 3000 units in two shifts.

Table 4.20 "New System 3000 Units Through Packing" – required number of runs.

Testing Area	3000 Units Through Packing						
	Average μ1	Min.	Max.	Conf. Int. Min.	Conf. Int. Max.	# Runs Mod	# Runs Est.
% Up-time Dielectric machine	39.6%	37.2%	42.8%	39.4%	39.8%	100.0	32.6
% Up-time Test. Mach. - Tester #1	67.4%	64.7%	71.1%	67.1%	67.6%	100.0	13.9
% Util. Test. Pers. – Tester #1	58.5%	56.1%	61.3%	58.3%	58.7%	100.0	11.1
% Up-time Draw Out tester	93.2%	89.6%	98.0%	92.8%	93.5%	100.0	13.6
% Utilization assembler	83.4%	81.5%	85.5%	83.3%	83.6%	100.0	2.6
% Utilization final inspector	71.6%	67.5%	77.5%	71.2%	72.0%	100.0	33.3
Units packed - Throughput	2620.3	2603.0	2637.0	2618.8	2621.8	100.0	0.1
% Utilization packers	100.0%	99.7%	100.0%	100.0%	100.0%	100.0	0.0
W.I.P.	271.9	198.2	370.6	264.9	279.0	100.0	6.6

Table 4.21 shows the simulation responses or output variables of interest for "3 Packers" model. It shows for each performance measure the expected average value, the minimum and maximum data observed, the confidence interval (minimum and maximum limits), the number of runs done during the simulation, and the minimum number of runs required for achieving the desired confidence interval accuracy for the average value. The conclusions were the following:

- No additional replicates were required.
- The results confirmed the criticality of the number of packers. Note that the packing personnel utilization was at 78.5% at an average of 3083.5 units packed.
- It was also concluded that a third shift was needed to achieve 3000 units monthly with the minimum capital investment on equipment. Please note that in Table 3.8 the number of testing machines was set to 6 per shift and the number of draw out machines was set to 3 per shift. Thus, suggesting the need to buy 4 additional testing machines and 2 additional draw out machines if

only two shifts were run. With a third shift, the additional testing machines and draw out machines were 2 and 1, respectively.

- The balancing issue between the testing and draw out/final inspection areas was still noticeable. At this stage, the question of balancing the labor was raised. The possibility for balancing was available by moving the "Inspection I" from the draw out/final inspection area to the testing area. This question must be answered in terms of labor cost. It was decided to add the number of testers as a critical factor, and to experiment further on in order to determine where to locate "Inspection I".

Table 4.21 "3 Packers" – required number of runs.

	3 Packers						
Testing Area	**Average μ2**	**Min.**	**Max.**	**Conf. Int. Min.**	**Conf. Int. Max.**	**# Runs Mod**	**# Runs Est.**
% Up-time Dielectric machine	39.59%	37.17%	42.84%	39.37%	39.82%	100.0	32.6
% Up-time Test. Mach. - Tester #1	67.40%	64.73%	71.08%	67.15%	67.65%	100.0	13.9
% Util. Test. Pers. – Tester #1	58.48%	56.13%	61.33%	58.28%	58.67%	100.0	11.1
% Up-time Draw Out tester	93.2%	89.6%	98.0%	92.8%	93.5%	100.0	13.6
% Utilization assembler	83.4%	81.5%	85.5%	83.3%	83.6%	100.0	2.6
% Utilization final inspector	71.6%	67.5%	77.5%	71.2%	72.0%	100.0	33.3
Units packed - Throughput	3083.5	2965.0	3228.0	3072.5	3094.5	100.0	2.0
% Utilization packers	78.5%	75.4%	82.1%	78.2%	78.8%	100.0	13.3
W.I.P.	20.9	16.8	41.2	20.3	21.6	100.0	10.8

Table 4.22 shows the results from the hypothesis tests. It shows for each performance measure of interest the mean difference between the "New System 3000 Units Through Packing" model and the "3 Packers" model, the confidence interval (minimum and maximum limits) and the suggested action from the tests. It was concluded that the model was sensitive to the number of packing personnel since H_0 was rejected for the performance measures related to the packing area. Nevertheless, it was decided to fix the number of packing personnel to 2 per shift, having 3 shifts. Note that

no significant difference was observed for the performance measures related to the testing and draw out/final inspection areas, but this was expected since the number of packing personnel had no effect on those areas. These results suggested the inclusion of this parameter (number of packing personnel) in further experimental designs.

Table 4.22 "New System 3000 Units Through Packing" vs. "3 Packers" – Hypothesis test

Testing Area	Hypothesis Testing Ho: $\mu 1 - \mu 2 = 0$; Ha: $\mu 1 - \mu 2 \neq 0$			
	Mean Diff.	**Conf. Int. Min.**	**Conf. Int. Max.**	**Comments**
% Up-time Dielectric machine	0.00%	0.00%	0.00%	Fail to Reject Ho
% Up-time Test. Mach. - Tester #1	0.00%	0.00%	0.00%	Fail to Reject Ho
% Util. Test. Pers. - Tester#1	0.00%	0.00%	0.00%	Fail to Reject Ho
% Up-time Draw Out tester	0.00%	0.00%	0.00%	Fail to Reject Ho
% Utilization assembler	0.00%	0.00%	0.00%	Fail to Reject Ho
% Utilization final inspector	0.00%	0.00%	0.00%	Fail to Reject Ho
Units packed - Throughput	-463	-474.1	-451.9	Reject Ho
% Utilization packers	21.50%	21.22%	21.78%	Reject Ho
W.I.P.	251	244.54	257.46	Reject Ho

4.2.2.5.Summary of the Sensitivity Analysis

The only parameter excluded from the critical or experimental factors list was the number of dielectric machines. The following input parameters were identified as critical factors: inter-arrival time, number of testing personnel, number of testing machines, testing process time, yield, labor balancing between testing area and draw out/final inspection area, number of assemblers, number of final inspectors, and number of draw out machines. In the case of the number of packing personnel, it was identified as a critical factor but a decision was made to fix the value to 2 per shift, having 3 shifts.

Having factors that could be classified as qualitative (categorical) and quantitative (both, discrete and continuous), it was very difficult at this stage to try any optimization

approach. It was decided to perform a series of experimental designs with the objective of reducing the number of critical factors. The initial focus was on the categorical factor labor balancing. The following section provides the results from those experiments.

4.3. Reducing the Problem with Experimental Designs

This section explains the results and conclusions related to the experimental designs performed with the main objective of reducing the number of critical factors or input parameters to be considered in the metamodels. A series of sequential experimental designs were executed with major focus on the categorical and discrete input parameters, in particular, the labor balancing issue, the number of assemblers, the number of final inspectors and the number of draw out machines.

The section has been divided in two sub-sections: (1) Labor Balancing, and (2) Setting Up the Draw Out/Final Inspection Area. Refer to sections 3.3.1 and 3.3.2 for more detail on the methodology.

4.3.1. Labor Balancing

The following are the results and conclusions related to the experimental design performed with respect to the labor balancing issue between the testing area and the draw out/final inspection area. As a reminder, the experiment was a 2^{8-2} fractional factorial design with 10 center points and 1 replicate, and design resolution V (refer to Appendix D).

It is important to mention that caution was taken at the time of doing the interpretation of the statistical results from this experiment. The main reason was that having only 1 replicate did not guarantee the true estimate of the error. So instead, it was decided for conclusions to rely on graphical plots rather than analysis of variance. As previously mentioned, this approach made sense since the focus was on the categorical factor labor balancing.

Even though all performance measures were statistically and graphically analyzed with respect to the experimental factors, only the graphical analysis related to the labor

cost per unit (refer to Equation 3.1 for the calculation of the labor cost) was included in this document. Conclusions related to the balancing issue were driven only by the analysis of the labor cost per unit performance metric. Other analyses with respect to the other output variables gave a flavor on the behavior of the system, which helped on defining the direction for future experimental designs but were not of value for the objective of this experiment.

Figure 4.14 is a plot of the main effects. It was observed that apparently the location of Inspection I (II) was of no major impact to the labor cost per unit as compared, for example to the inter-arrival time (IT), the number of final inspectors (FI), the number of testing personnel (TP), the number of assembly personnel (AP) and the yield (Y). It is important to clarify that TM stands for the number of testing machines and TPT stands for testing process time. The effect of these factors on the labor cost per unit was somehow predictable, but they were a reminder that at some point in time, the simulation study would have to consider the optimization. For example, if more units get into the system and the yield increases, there is no doubt that the labor cost per unit will decrease (assuming everything else stays the same), but these actions will require the addition of personnel which in turn will tend to increase the labor cost per unit. The question is then raised about where is the point in which the benefit of increasing the arrival rate is maximized with respect to the interrelation within all factors and the performance measures, considering the constraints defined by the management of the company.

Figure 4.15 is a box plot of the line balancing factor. It was again observed that apparently the location of Inspection I (II) was of no major impact to the labor cost per unit, not only in terms of the mean value but also in terms of the spread of the data. As in Figure 4.14, there was a slight decrease in cost when Inspection I was located at the draw out/final inspection area.

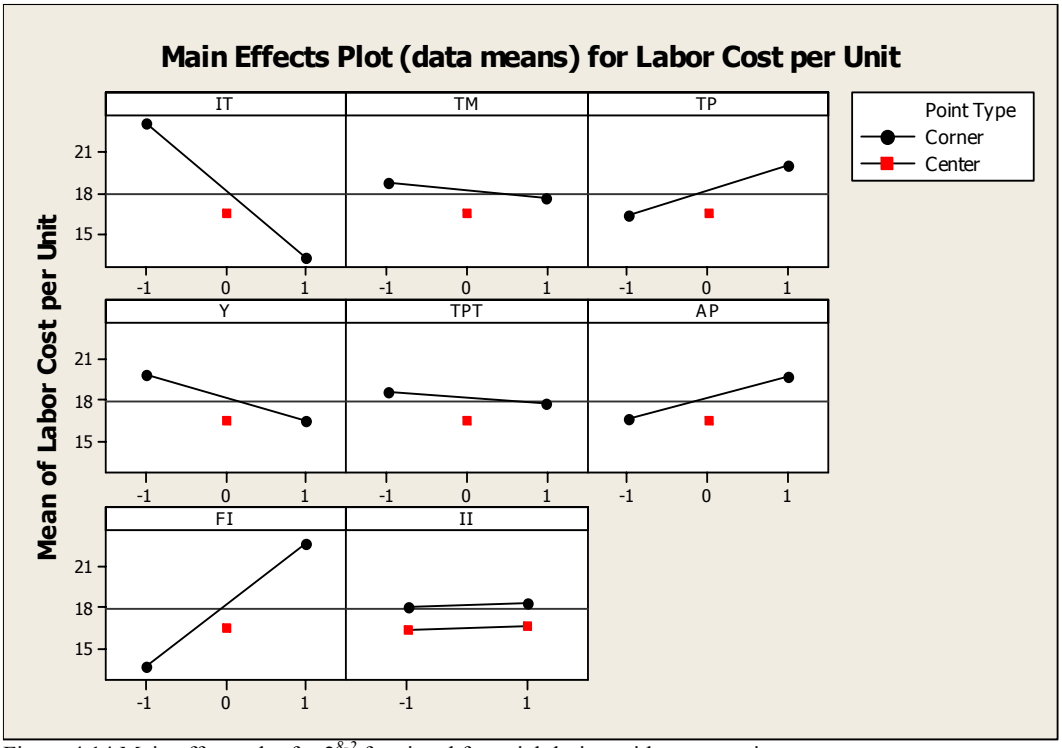

Figure 4.14 Main effects plot for 2^{8-2} fractional factorial design with center points.

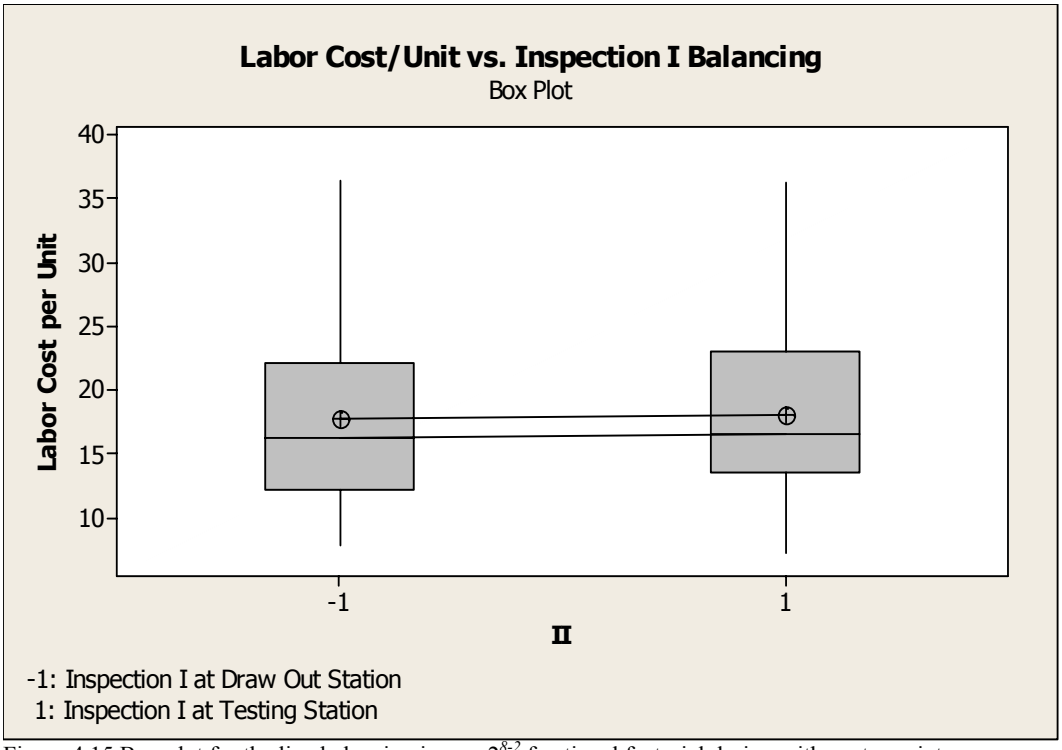

Figure 4.15 Box plot for the line balancing issue - 2^{8-2} fractional factorial design with center points.

It could be concluded from Figures 4.14 and 4.15 that locating Inspection I at the draw out/inspection area was slightly beneficial as compared to the test area. Nevertheless, these two charts were not providing information with respect to the possible combinations that could be done with the number of final inspectors (FI), assemblers (AP) and testing personnel (TP). The question to answer before making a final decision was if there was a combination of these factors in which it was preferable to locate Inspection I at the testing area. This information was provided by Figure 4.16. Note that independently of the level at which the balancing factor was set, the pattern was similar. It was then decided not to balance the labor and to maintain Inspection I at the draw out/final inspection area. As a result, the number of assembly personnel and the number of final inspection personnel were not considered anymore as critical factors. But before doing so, it was necessary to define the fix values at which to set these parameters.

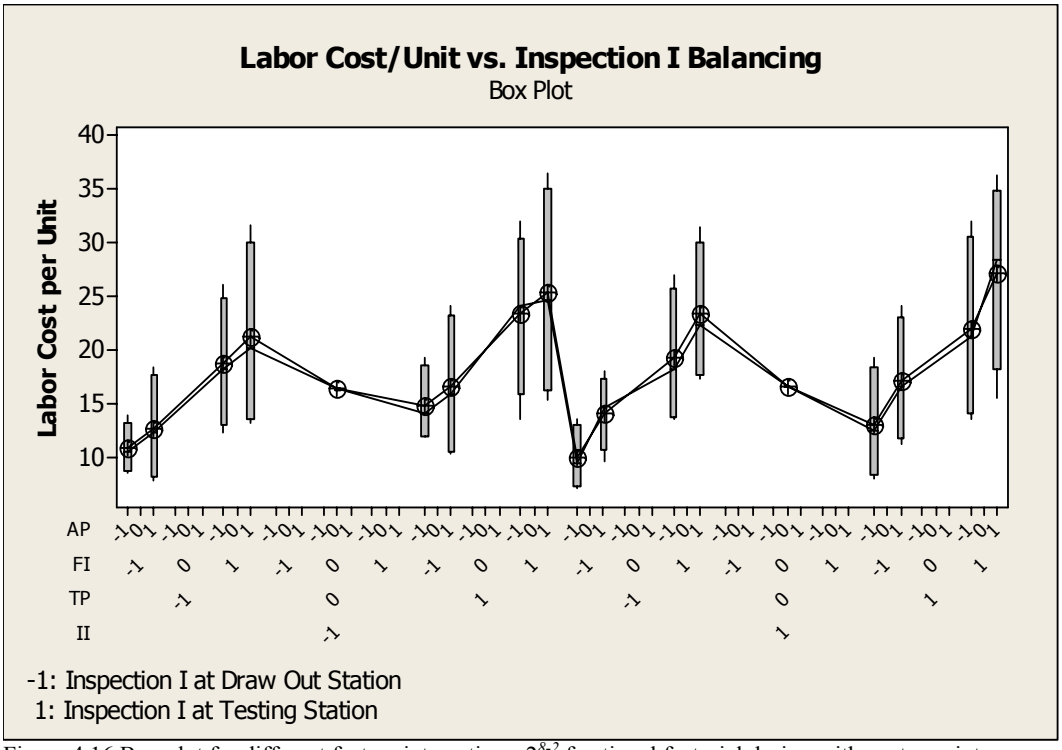

Figure 4.16 Box plot for different factors interaction - 2^{8-2} fractional factorial design with center points.

Figure 4.17 provided interesting information with respect to the number of draw out machines. Note that independently of the level of the balancing factor Inspection I (II), the box plot showed up-time utilization above 95%, which was the limit specified by

the management of the company. This meant that as with the number of assembly and final inspection personnel, it was necessary to define at which level to set the number of draw out machines.

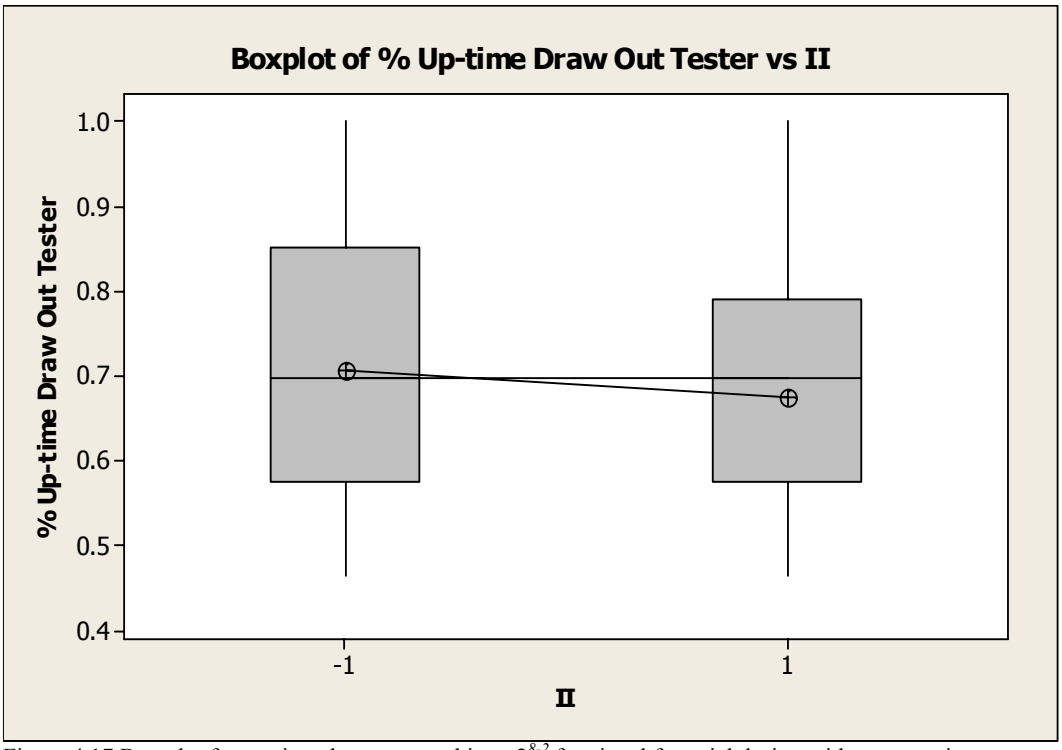

Figure 4.17 Box plot for up-time draw out machine - 2^{8-2} fractional factorial design with center points.

In summary, it was decided to leave Inspection I at the draw out/final inspection area, and to define a value at which to fix the number of assembly and final inspection personnel, as well as the number of draw out machines by means of a 2^k factorial design. Results of this experiment are explained in the next section.

4.3.2. Setting Up the Draw Out/Final Inspection Area

The following are the results and conclusions related to the experimental design performed with respect to the draw out/final inspection area. The main objective of this experiment was to define a set of values at which to fix the number of assembly personnel, the number of final inspection personnel, and the number of draw out machines. As a reminder, the experiment was a 2^3 factorial design with 3 replicates, 3 blocks, and a total of 24 design points (refer to Appendix E).

The performance measures of interest were the labor cost per unit, the labor utilization for the assembly and the final inspection personnel, the up-time for the draw out machine, and the work-in-process inventory, in particular, the buffer in front of the draw out machine. These metrics were statistically and graphically analyzed, but the explanation of the results focused on the graphical analysis since the experimental factors were discrete variables. As in the previous experimental design, the whole package of analyses improved the understanding of the model behavior, but only the analyses pertinent to the objectives of this experiment were explained in this section.

Table 4.23 Data sheet for 2^3 factorial design.

Run Order	Experimental Treatments			Performance Measures						
	Assy. Pers.	Final Insp.	D. O. Mach.	% Up-time D. O. Mach.	% Util. Assy. Pers.	% Util. Final Insp.	Labor Cost per Unit	WIP (Avg.)	Buffer D. O. (Max.)	Buffer D. O. (Avg.)
1	-1	1	1	0.677	0.811	0.661	$7.51	12.1	0.5	0.3
2	-1	1	-1	0.986	0.779	0.687	$7.53	36.7	95.5	26.6
3	-1	-1	-1	0.992	0.871	0.884	$6.05	46.6	111.1	35.4
4	-1	-1	1	0.726	0.899	0.861	$6.11	14.0	0.3	0.2
5	1	-1	1	0.676	0.765	0.680	$7.30	12.4	0.4	0.3
6	1	1	-1	0.985	0.708	0.523	$8.67	35.1	83.2	25.1
7	1	1	1	0.664	0.723	0.509	$8.57	11.9	0.6	0.4
8	1	-1	-1	0.987	0.742	0.715	$7.07	37.9	88.1	27.6
9	-1	-1	1	0.724	0.897	0.858	$6.13	13.8	0.3	0.2
10	1	-1	-1	0.985	0.741	0.712	$7.09	37.9	94.3	27.5
11	-1	1	1	0.675	0.809	0.658	$7.46	12.1	0.4	0.3
12	1	-1	1	0.671	0.761	0.676	$7.13	12.3	0.4	0.3
13	-1	1	-1	0.984	0.778	0.683	$7.49	35.8	137.2	25.7
14	1	1	-1	0.984	0.707	0.523	$8.35	34.2	74.8	24.2
15	1	1	1	0.659	0.720	0.504	$8.56	11.8	0.7	0.4
16	-1	-1	-1	0.990	0.870	0.883	$6.06	46.3	124.0	35.2
17	-1	1	1	0.679	0.812	0.663	$7.39	12.2	0.5	0.3
18	1	-1	-1	0.987	0.742	0.716	$7.13	37.7	93.5	27.4
19	1	1	1	0.664	0.723	0.510	$8.54	11.9	0.6	0.4
20	1	-1	1	0.676	0.765	0.681	$7.12	12.4	0.4	0.3
21	-1	1	-1	0.987	0.780	0.687	$7.39	37.4	78.2	27.2
22	-1	-1	1	0.726	0.900	0.862	$6.02	14.0	0.3	0.2
23	-1	-1	-1	0.992	0.871	0.885	$5.95	47.7	127.8	36.6
24	1	1	-1	0.986	0.708	0.525	$8.52	35.9	76.1	26.0

Table 4.23 shows the data obtained from the simulation output for each performance measure of interest. Note that the utilization for the assembly personnel was never above 90%, which was the limit specified by the management of the company. The average work-in-process inventory was explained by the buffer in front of the draw out machine, meaning that at some particular design points, the draw out machine became the bottleneck. Looking into more detail, the average work-in-process inventory exceeded the limit of 22 units only when the utilization of the draw out machine was higher than the 95% limit. It was decided from these observations that the variables of major interest for the analysis were the labor cost per unit, the utilization of the final inspection personnel, and the up-time of the draw out machine.

Figures 4.18 and 4.19 demonstrated that the significant factors for the labor cost per unit were the number of assembly personnel (AP) and the number of final inspection personnel (FI). Neither the number of draw out machines (DOM) nor interactions of the factors were critical.

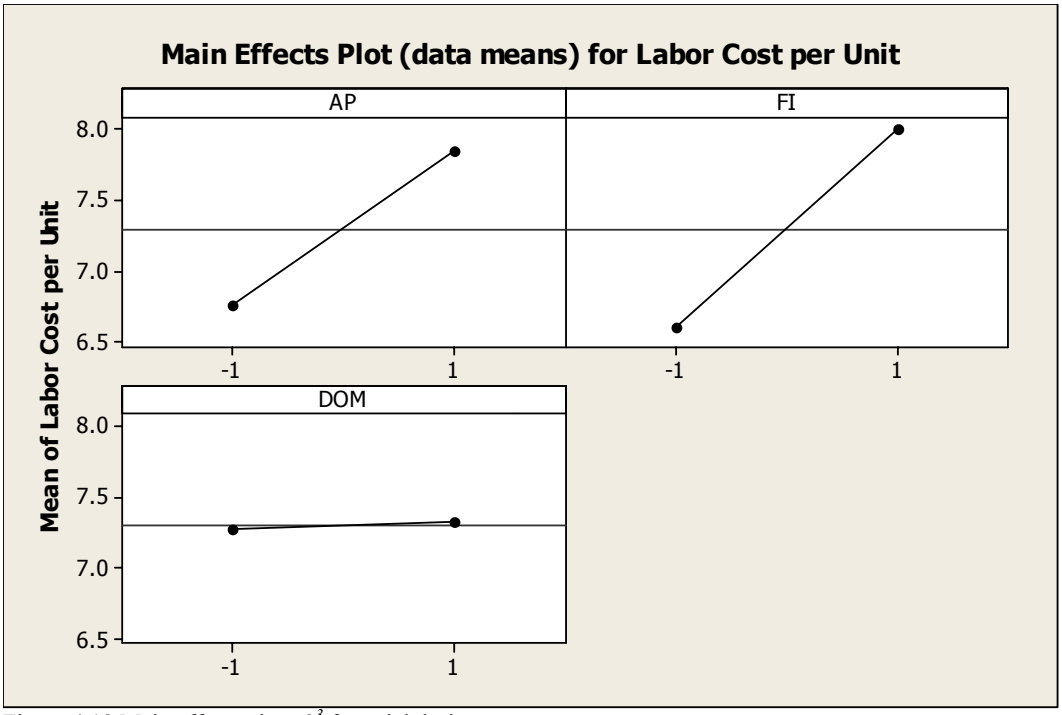

Figure 4.18 Main effects plot - 2^3 factorial design.

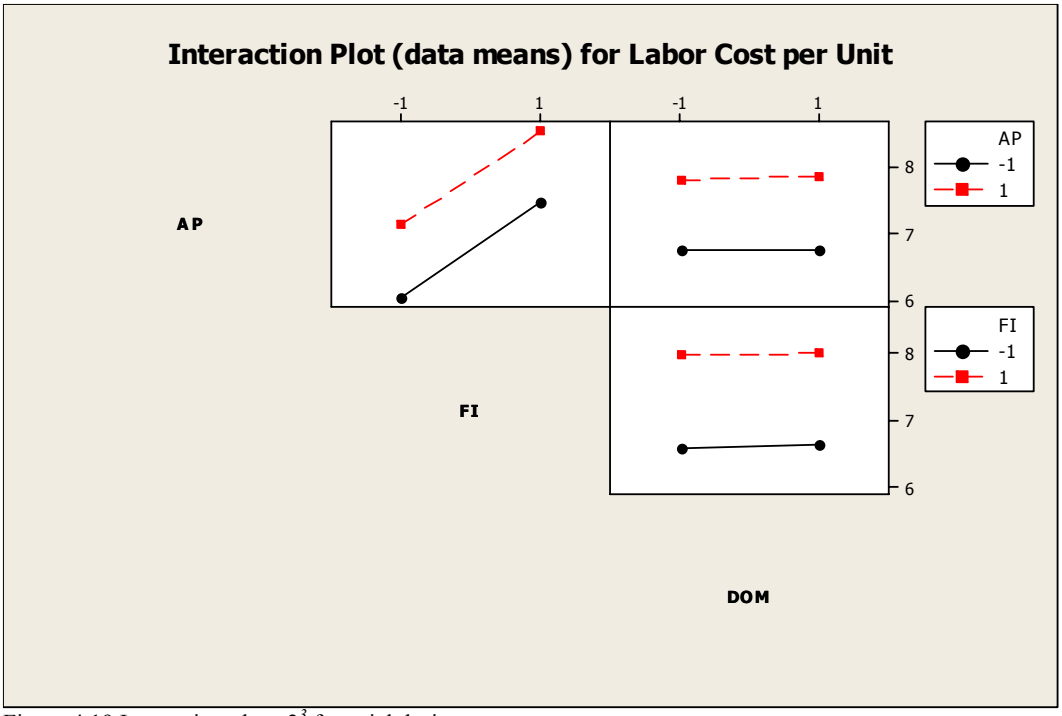

Figure 4.19 Interaction plot - 2^3 factorial design.

Figure 4.20 suggested that the number of assembly personnel (AP) and final inspection personnel (FI) should be set at the minimum levels. Nevertheless, that alternative was not an option since as per Figure 4.21, the utilization of the final inspector was above the limit of 80% for all design points calling for that combination. Setting the number of final inspection personnel to the lower experimental level, and the number of assembly personnel to the higher level was a feasible option.

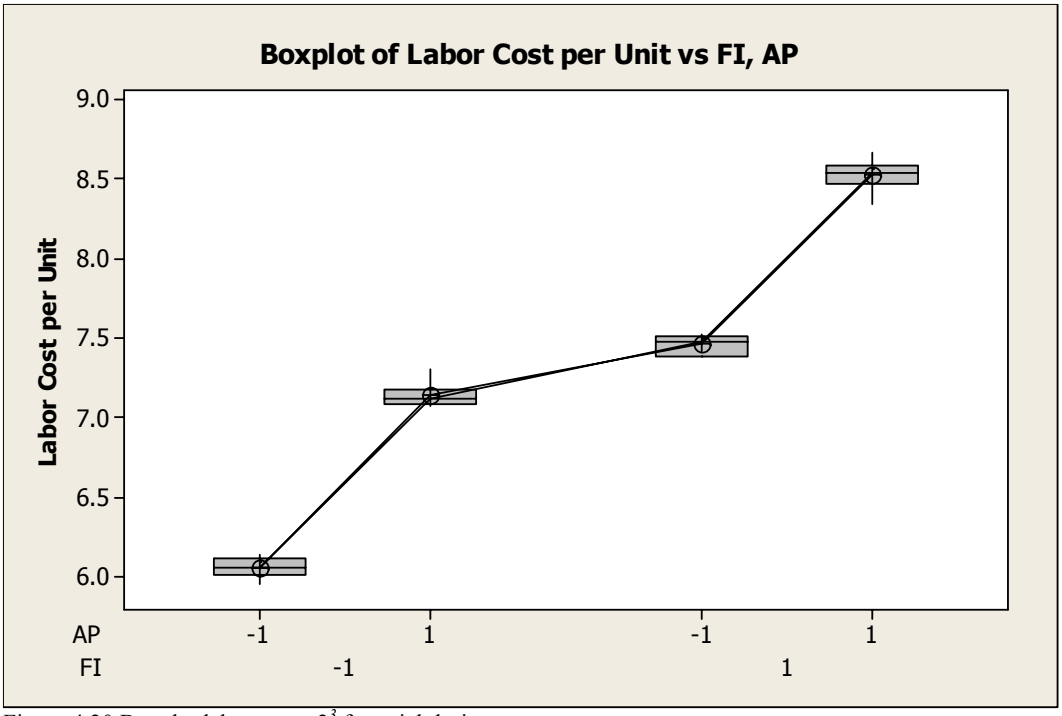

Figure 4.20 Boxplot labor cost - 2^3 factorial design.

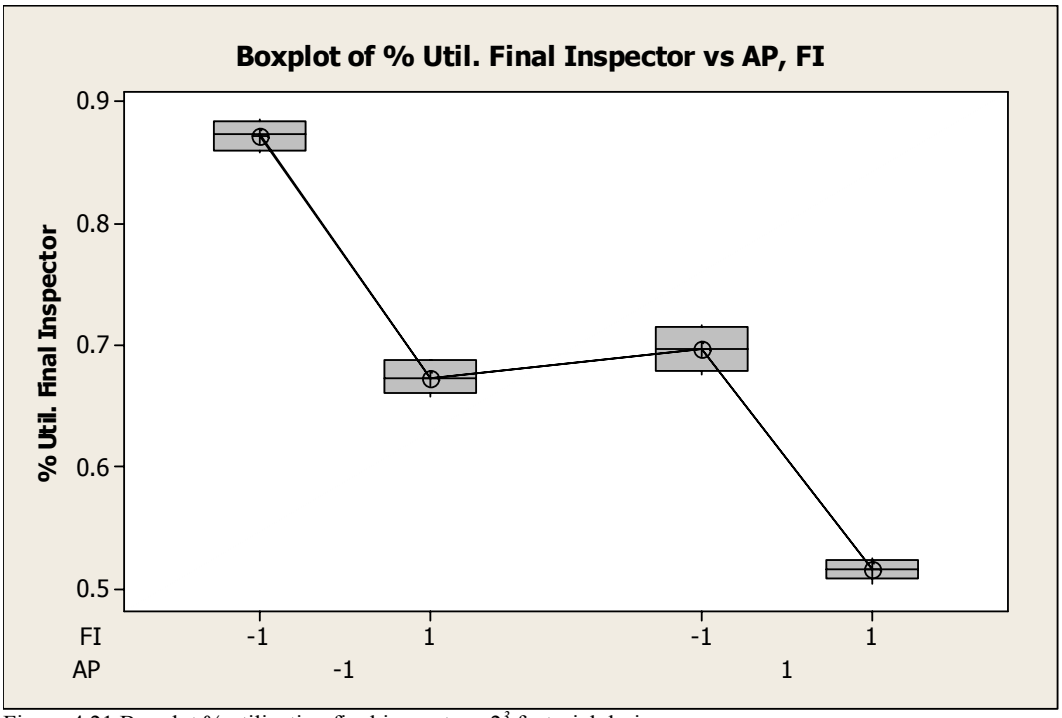

Figure 4.21 Boxplot % utilization final inspector - 2^3 factorial design

Figure 4.22 showed that the number of draw out machines (DOM) should be set at the higher level, irrespective of the combination between the number of assembly

personnel (AP) and final inspection personnel (FI). Going back to Table 4.23, it was observed that the work-in-process inventory was always less than 22 units when the number of draw out machines was set to the higher level.

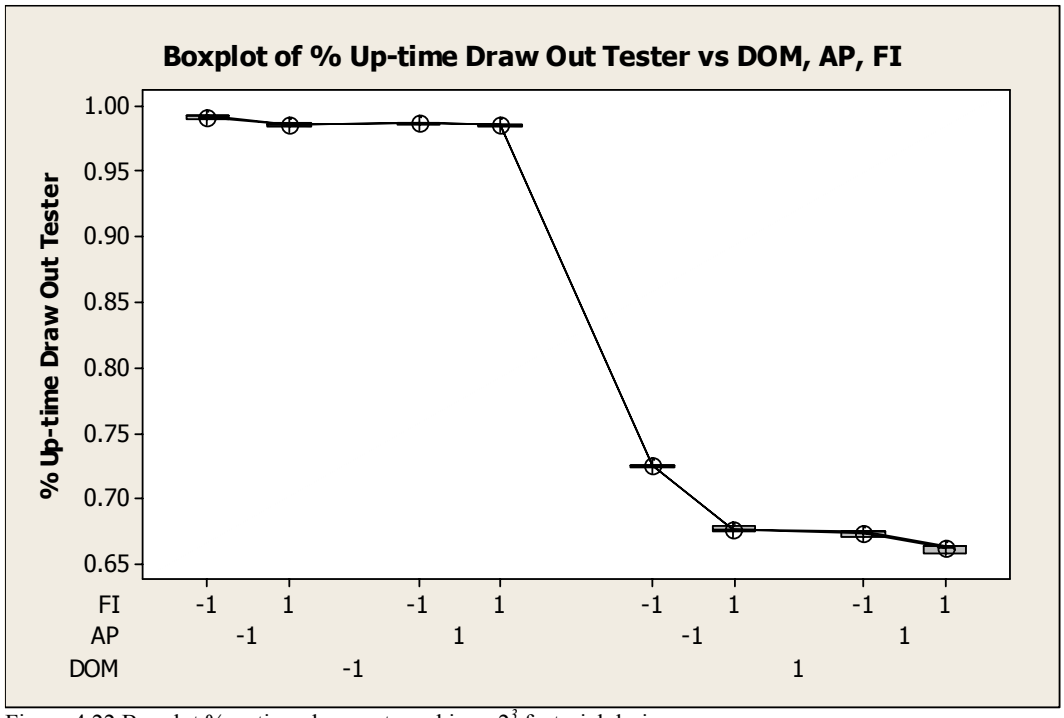

Figure 4.22 Boxplot %up-time draw out machine - 2^3 factorial design.

In summary, the conclusion of this experiment was to fix the number of assembly personnel, the number of final inspection personnel, and the number of draw out machines to 4, 2, and 3, respectively. These will be the levels at which to fix these input parameters for future simulation runs related to the experimental design phase.

4.4. Development of Metamodels

This section explains the results and conclusions of the experimental designs performed in order to develop the metamodels that were used in the optimization process of the study for the testing area. In total, three parallel central composite experiments were designed; one experiment for each testing machine.

The section has been divided in three sub-sections: (1) Metamodels for 3 Testing Machines, (2) Metamodels for 4 Testing Machines, and (3) Metamodels for 5 Testing Machines. Whenever appropriate, the section refers to the Appendixes for backup

information or data. Refer to sections 3.4.1, 3.4.2, and 3.4.3 for more details on the methodology.

4.4.1. Metamodels for 3 Testing Machines

This section covers the results and conclusions related to the development of metamodels when 3 testing machines were used. As previously mentioned, the experiment was a central composite design with 3 factors and two levels at each factor, 2 replicates and 2 blocks.

The performance measures of interest were the throughput, the up-time for the testing machines, and the work-in-process inventory. The following analysis was done for each of these response variables: regression analysis, analysis of variance including the lack of fit test, normal probability plot for residuals, histogram of the residuals, plot of the residual vs. fitted values and plot of the residuals vs. the order of the data.

Table 4.24 presents the data obtained from the simulation output for each performance measure of interest. Note that the utilization of the assembly personnel, the final inspector and the packing personnel was never above the respective limits specified by the management of the company. In addition, the up-time for the draw out machine did not exceed the specified limit. For this reason, it was decided not to build metamodels for those variables.

After observing the throughput, minimum and average columns, it was decided to use the data of the average to build the metamodel. It was also decided to build metamodels for the up-time of the testing machines and the work-in-process inventory, since their respective limits were exceeded for some experimental runs.

Table 4.24 Data sheet for 2^3 central composite factorial design – 3 testing machines.

	Up-Time Testing Machine	Up-Time Draw Out Machine	Util. Assy. Pers.	Util. Final Insp.	Through (Min.)	Through (Avg.)	Util. Packing Pers.	WIP
Run Order	Model	Model	Model	Model	Model	Model	Model	Model
1	0.956	0.632	0.752	0.600	956	1046.2	0.799	20.5
2	0.947	0.648	0.763	0.628	997	1073.4	0.819	19.9
3	0.959	0.634	0.754	0.602	985	1051.4	0.802	22.5
4	0.945	0.626	0.745	0.594	963	1037.4	0.791	19.4
5	0.927	0.635	0.751	0.610	961	1050.1	0.801	16.8
6	0.979	0.625	0.749	0.586	958	1036.6	0.791	32.5
7	0.973	0.644	0.763	0.617	990	1065.9	0.814	28.0
8	0.929	0.633	0.749	0.608	977	1047.3	0.799	16.6
9	0.951	0.630	0.749	0.597	975	1041.6	0.794	20.2
10	0.956	0.651	0.767	0.630	1001	1075.1	0.821	21.6
11	0.952	0.630	0.750	0.597	919	1041.9	0.795	20.3
12	0.959	0.612	0.737	0.567	940	1016.7	0.776	21.1
13	0.957	0.614	0.737	0.572	930	1017.0	0.777	20.9
14	0.950	0.629	0.748	0.596	927	1040.9	0.794	19.7
15	0.957	0.632	0.752	0.600	963	1046.4	0.798	21.4
16	0.953	0.629	0.749	0.597	952	1041.1	0.794	20.4
17	0.979	0.648	0.767	0.622	1005	1072.8	0.819	34.3
18	0.979	0.629	0.752	0.590	961	1041.1	0.794	31.9
19	0.932	0.616	0.737	0.578	946	1020.2	0.779	16.5
20	0.928	0.615	0.734	0.577	931	1017.9	0.777	16.2
21	0.944	0.624	0.744	0.589	950	1034.3	0.790	18.6
22	0.955	0.631	0.752	0.598	972	1044.9	0.797	21.2
23	0.956	0.610	0.735	0.562	943	1011.7	0.772	20.4
24	0.971	0.641	0.760	0.613	985	1060.9	0.810	26.9
25	0.956	0.631	0.752	0.597	972	1045.1	0.798	20.8
26	0.951	0.648	0.764	0.625	1001	1072.0	0.818	20.5
27	0.952	0.628	0.748	0.594	927	1040.6	0.794	20.9
28	0.924	0.631	0.749	0.603	968	1045.3	0.797	16.0
29	0.921	0.629	0.745	0.601	973	1040.3	0.794	15.8
30	0.951	0.610	0.733	0.566	947	1012.5	0.773	19.9
31	0.924	0.610	0.731	0.568	932	1010.6	0.771	15.6
32	0.947	0.627	0.747	0.593	962	1037.9	0.792	18.8
33	0.950	0.629	0.749	0.597	953	1041.6	0.795	20.3
34	0.976	0.644	0.764	0.616	980	1065.7	0.814	30.4
35	0.944	0.647	0.762	0.626	973	1070.0	0.816	19.0
36	0.947	0.625	0.745	0.590	962	1035.0	0.790	19.2
37	0.950	0.630	0.749	0.596	977	1041.3	0.794	20.2
38	0.976	0.621	0.746	0.579	950	1030.6	0.787	30.1
39	0.977	0.626	0.751	0.586	962	1038.9	0.793	31.5
40	0.932	0.616	0.736	0.578	935	1020.4	0.780	16.6

4.4.1.1.Throughput

Figure 4.23 is an output summary from Minitab for the coefficients of the regression and the analysis of variance.

Response Surface Regression: Throughput (Avg.) versus Block, IT, Y, TPT

The analysis was done using coded units.

Estimated Regression Coefficients for Throughput (Avg.)

Term	Coef	SE Coef	T	P
Constant	1041.64	0.6831	1524.792	0.000
Block	-2.04	0.3745	-5.433	0.000
IT	11.60	0.4532	25.604	0.000
Y	17.42	0.4532	38.432	0.000
TPT	0.78	0.4532	1.711	0.098
IT*IT	-0.10	0.4412	-0.217	0.830
Y*Y	0.72	0.4412	1.626	0.115
TPT*TPT	-0.08	0.4412	-0.177	0.861
IT*Y	0.50	0.5922	0.844	0.405
IT*TPT	0.65	0.5922	1.098	0.281
Y*TPT	-0.05	0.5922	-0.084	0.933

S = 2.369 R-Sq = 98.7% R-Sq(adj) = 98.2%

Analysis of Variance for Throughput (Avg.)

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Blocks	1	165.6	165.6	165.65	29.52	0.000
Regression	9	12009.4	12009.4	1334.38	237.81	0.000
Linear	3	11982.4	11982.4	3994.14	711.83	0.000
Square	3	16.2	16.2	5.39	0.96	0.424
Interaction	3	10.8	10.8	3.60	0.64	0.594
Residual Error	29	162.7	162.7	5.61		
Lack-of-Fit	19	72.0	72.0	3.79	0.42	0.951
Pure Error	10	90.7	90.7	9.07		
Total	39	12337.8				

Figure 4.23 Regression analysis for throughput – 3 testing machines.

The analysis of the coefficients suggested that the model had a linear behavior since only main factors (inter-arrival time and yield) were identified as critical elements of the model. Note that the block element was also considered as critical; this is as a result of using new set of seeds (for the random generators) for each new replicate and having only two replicates at the cube points and axial points. The impact of the seeds could have been minimized with more replicates. Nevertheless, the coefficient of the block was not included in the metamodel Equation (4.1), thus going into the error of the throughput estimate.

$$Throughput_{3TM} = 1041.64 + 11.60 * IT + 17.42 * Y \quad (4.1)$$

Where, $Throughput_{3TM}$ is the estimate for the throughput when 3 testing machines are used, IT is the inter-arrival time, and Y is the yield.

It was confirmed by the analysis of variance that the model had a linear behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the throughput. The assumptions of normality and randomness of the residuals related to the analysis of variance were validated. Figure 4.24 presents several plots related to the normality and randomness of the residuals, such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals, residuals vs. the order of the data. These plots suggested the validation of the assumptions, but in order to have a robust conclusion, a normality test and runs test were done. Figure 4.25 and Figure 4.26 shows the results of those tests. The normality assumption was clearly validated from the normality test (observe $p > 0.05$). As for the runs test, all *p values* from the hypothesis were greater than $\alpha = 0.05$ except for the oscillation test. Oscillation is observable at the second half of the chart and was attributed to the seeds of the random number generators. It was decided to accept with no major concern the randomness of the residuals based on the results of the other runs test.

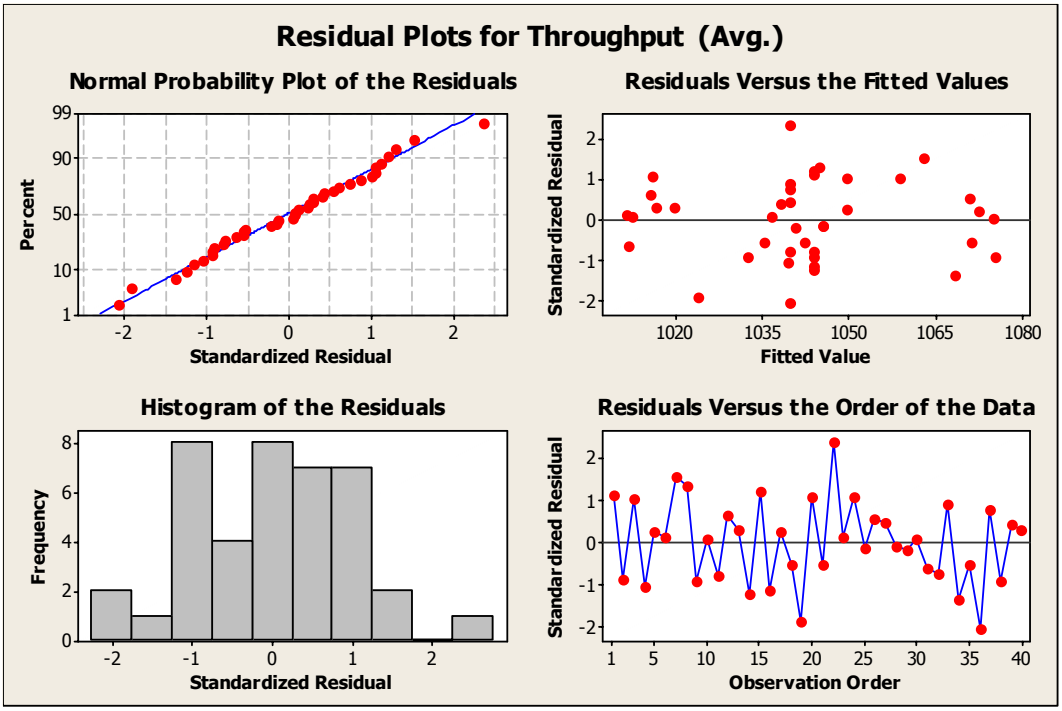

Figure 4.24 Residual plots for throughput – 3 testing machines.

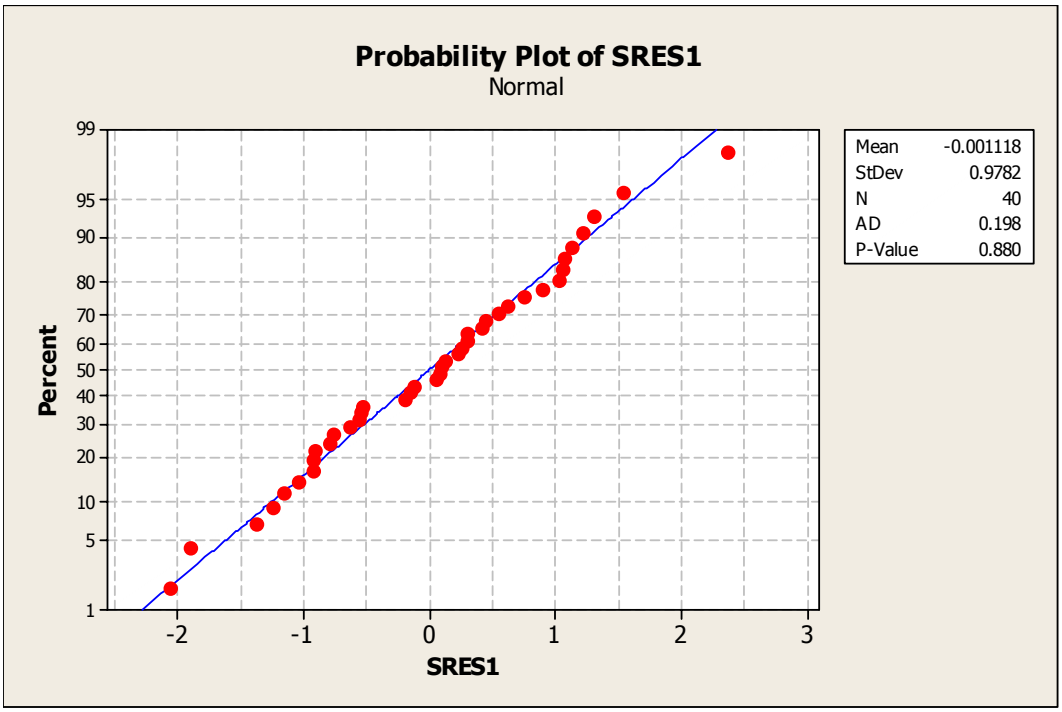

Figure 4.25 Normal probability plot for throughput – 3 testing machines.

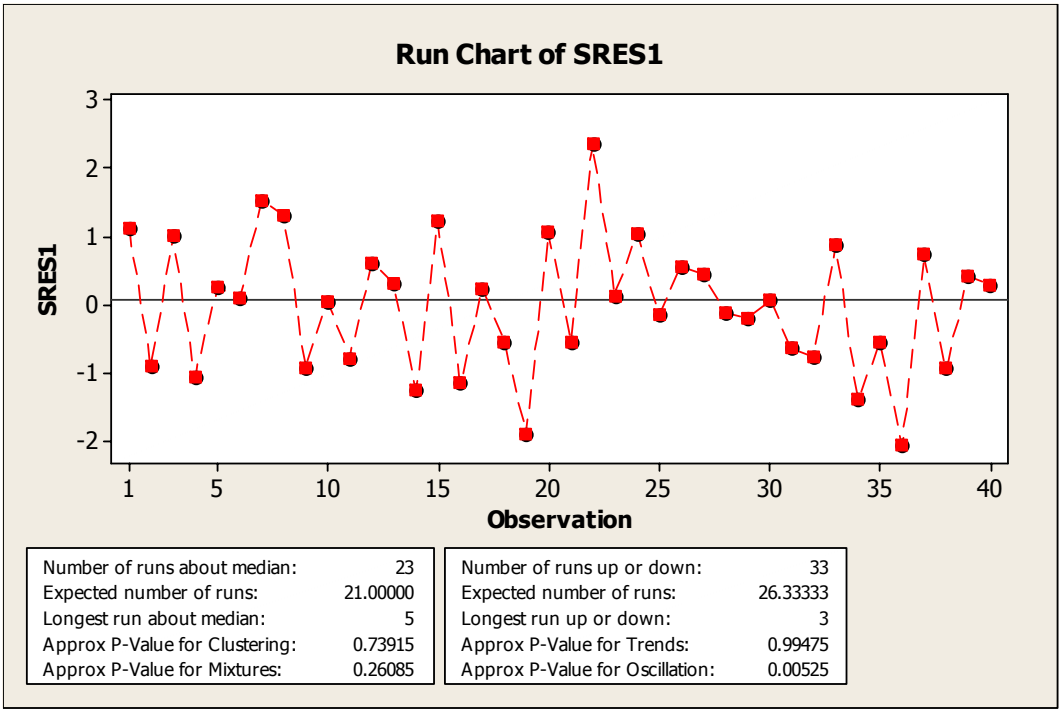

Figure 4.26 Runs test for throughput – 3 testing machines.

Figure 4.27 is a surface plot for the critical factors (inter-arrival time and yield); note the linear relation with respect to the throughput. It was observed that the throughput increases as both factors moved toward their respective high level on the experimental region. Note that yield was restricted at its high level of the experimental region based on the 90% constraint specified by the management of the company.

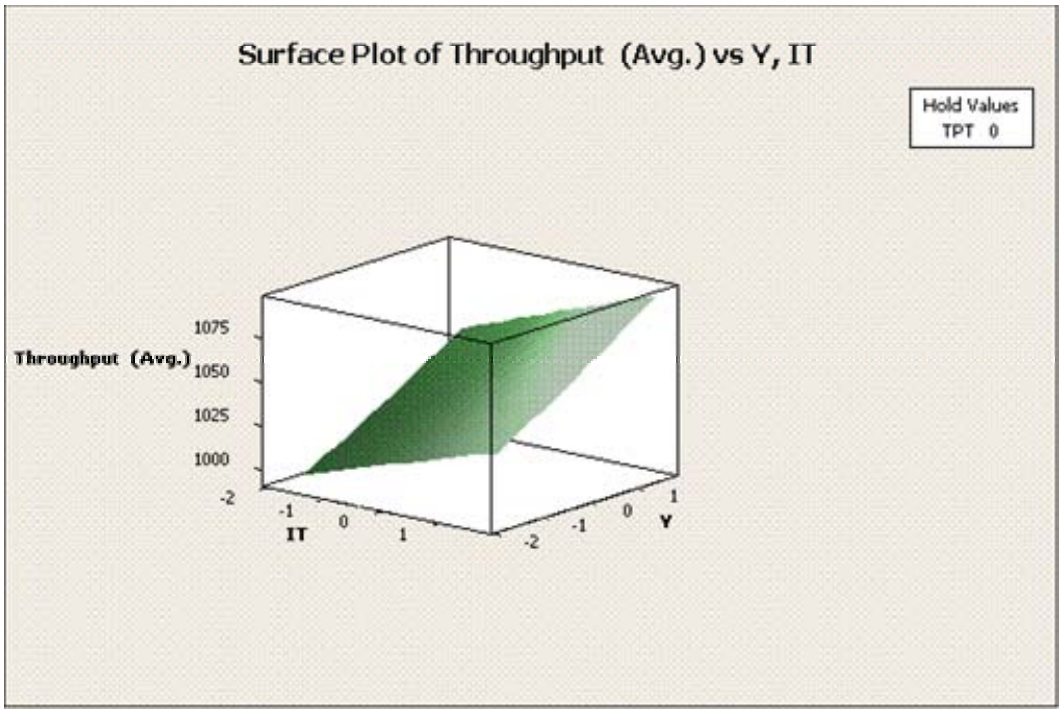

Figure 4.27 Surface plot for throughput vs. inter-arrival time and yield – 3 testing machines.

4.4.1.2.Up-time Testing Machines

Figure 4.28 is an output summary from Minitab for the coefficients of the regression and the analysis of variance. The analysis of the coefficients suggested that the model had a linear behavior since only main factors (inter-arrival time and testing process time) were identified as critical elements of the model. Note that the block element was also considered as critical; this is as a result of using new set of seeds (for the random generators) for each new replicate and having only two replicates at the cube points and axial points. The impact of the seeds could have been minimized with more replicates. Nevertheless, the coefficient of the block was not included in the metamodel Equation (4.2), thus going into the error of the up-time estimate.

$$Up-timeTestingMachines_{3TM} = 0.9517 + 0.010674 * IT - 0.015936 * TPT \qquad (4.2)$$

Where, *Up-timeTesting Machines*$_{3TM}$ is the estimate for the testing machines up-time when 3 testing machines are used, *IT* is the inter-arrival time, and *TPT* is the testing process time.

```
The analysis was done using coded units.

Estimated Regression Coefficients for Up-Time Testing Machine

Term           Coef   SE Coef         T      P
Constant   0.951700  0.000654  1455.328  0.000
Block     -0.001600  0.000359    -4.463  0.000
IT         0.010674  0.000434    24.601  0.000
Y         -0.000025  0.000434    -0.058  0.955
TPT       -0.015936  0.000434   -36.729  0.000
IT*IT     -0.000098  0.000422    -0.233  0.817
Y*Y        0.000520  0.000422     1.232  0.228
TPT*TPT   -0.000275  0.000422    -0.651  0.520
IT*Y       0.000188  0.000567     0.331  0.743
IT*TPT    -0.000188  0.000567    -0.331  0.743
Y*TPT      0.000062  0.000567     0.110  0.913

S = 0.002268   R-Sq = 98.6%   R-Sq(adj) = 98.1%


Analysis of Variance for Up-Time Testing Machine

Source          DF    Seq SS    Adj SS    Adj MS       F      P
Blocks           1  0.000102  0.000102  0.000102   19.92  0.000
Regression       9  0.010061  0.010061  0.001118  217.41  0.000
  Linear         3  0.010048  0.010048  0.003349  651.42  0.000
  Square         3  0.000011  0.000011  0.000004    0.74  0.540
  Interaction    3  0.000001  0.000001  0.000000    0.08  0.972
Residual Error  29  0.000149  0.000149  0.000005
  Lack-of-Fit   19  0.000063  0.000063  0.000003    0.39  0.963
  Pure Error    10  0.000086  0.000086  0.000009
Total           39  0.010312
```

Figure 4.28 Regression analysis for Uptime testing machine – 3 testing machines.

It was confirmed by the analysis of variance that the model had a linear behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the up-time for the testing machines. The assumptions of normality and randomness of the residuals related to the analysis of variance were validated. Figure 4.29 presents several plots related to the normality and randomness of the residuals, such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals, residuals vs. the order of the data. These plots suggested the validation of the assumptions, but in order to have a robust conclusion, a normality test and runs test were done. Figure 4.30 and Figure 4.31 shows the results of those tests. The normality assumption was clearly validated from the normality test (observe $p > 0.05$). As for the runs test, all *p values* from the hypothesis were greater than $\alpha = 0.05$ except for the oscillation test. Oscillation is observable at the second half of the chart and was attributed to the seeds of the random number generators. It was decided to

accept with no major concern the randomness of the residuals based on the results of the other runs test.

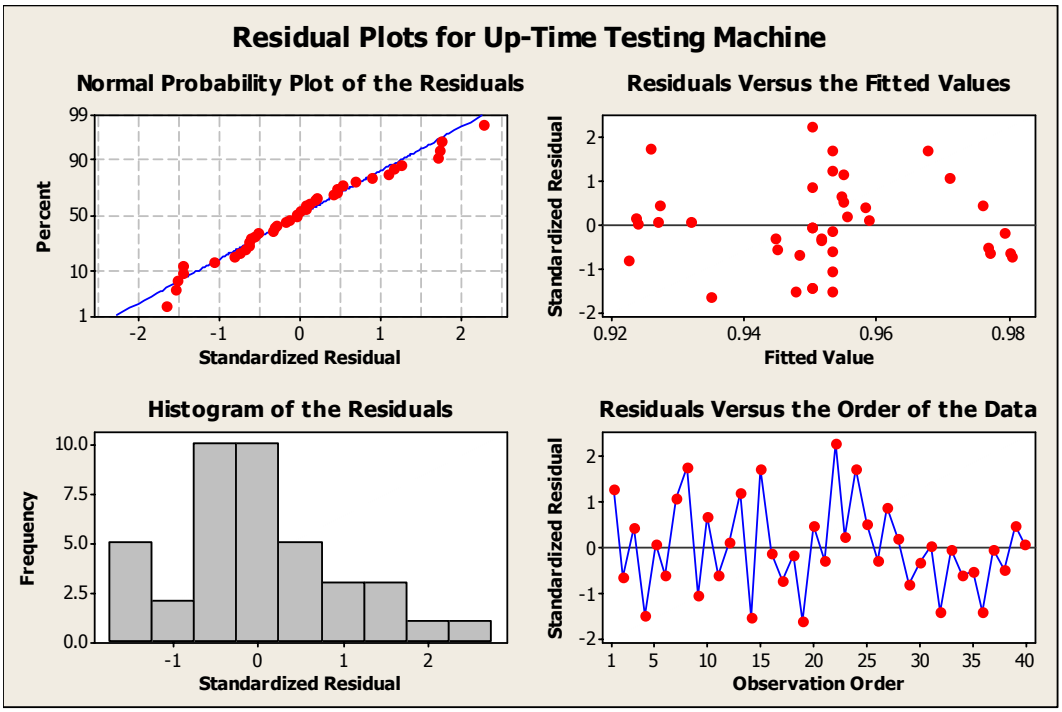

Figure 4.29 Residual plots for Uptime testing machine – 3 testing machines.

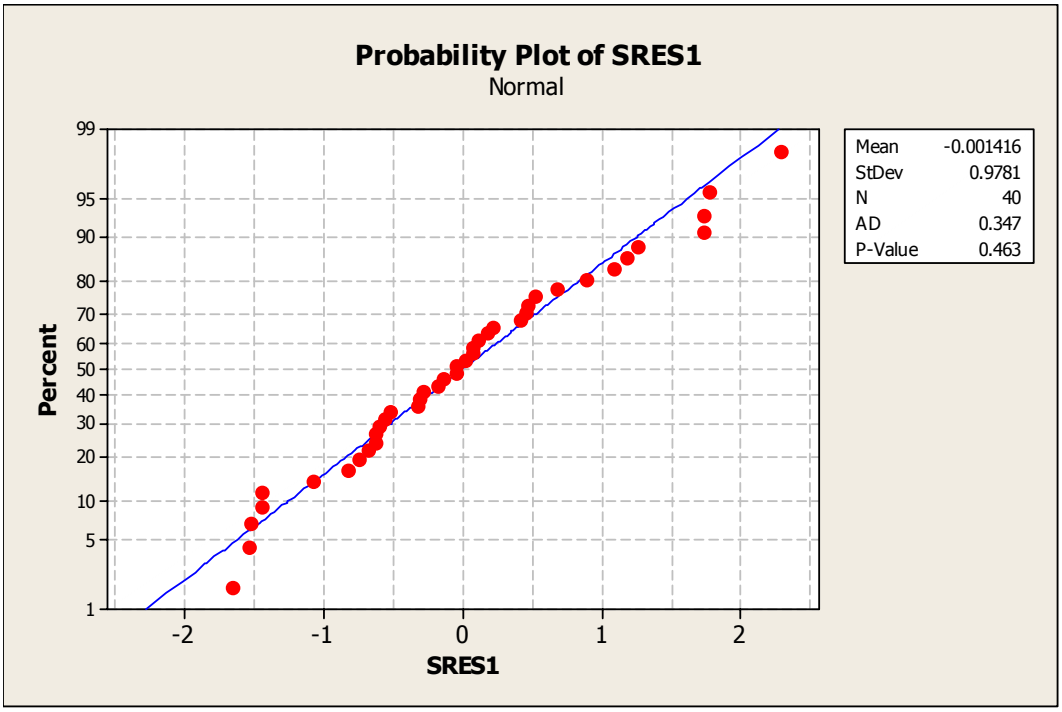

Figure 4.30 Normal probability plot for Uptime testing machine – 3 testing machines.

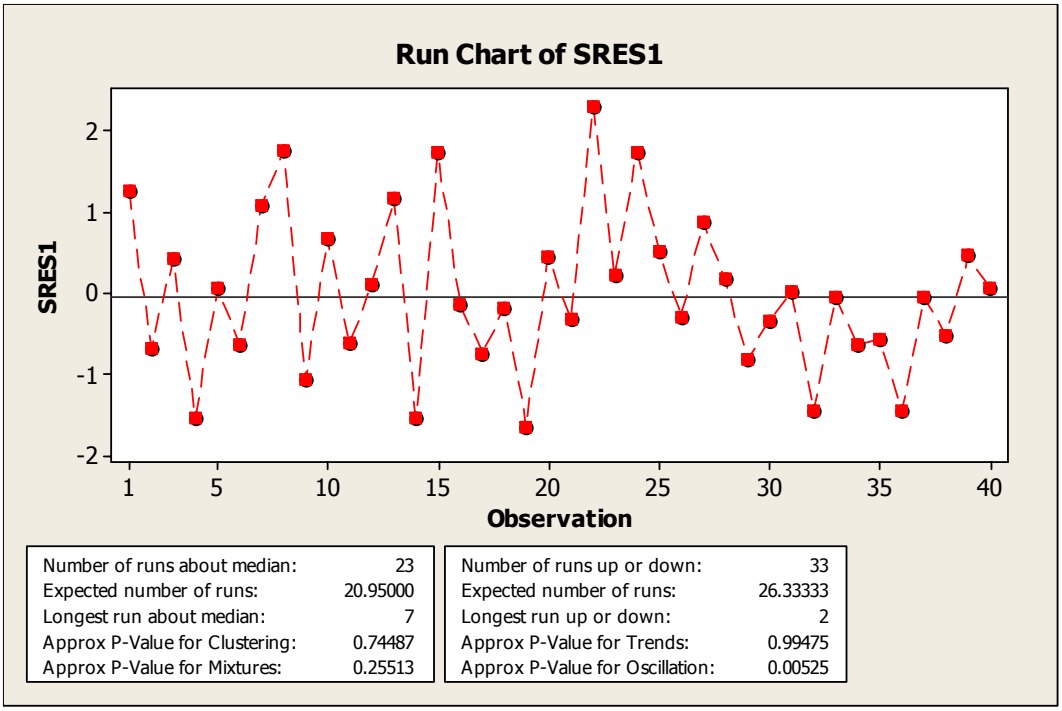

Figure 4.31 Runs test for Uptime testing machine – 3 testing machines.

Figure 4.32 is a surface plot for the critical factors (inter-arrival time and testing process time); note the linear relation with respect to the up-time of the testing machines. Note that as the testing process time approaches its lower level of the experimental region, the up-time for the testing machine is above the 95% constraint. Thus, it was concluded that the testing process time factor was restricted at both sides of the experimental region since the high level was the limit given by the management of the company. It was also concluded that the inter-arrival factor was restricted by its high level of the experimental region based on the restriction for the up-time of the testing machines.

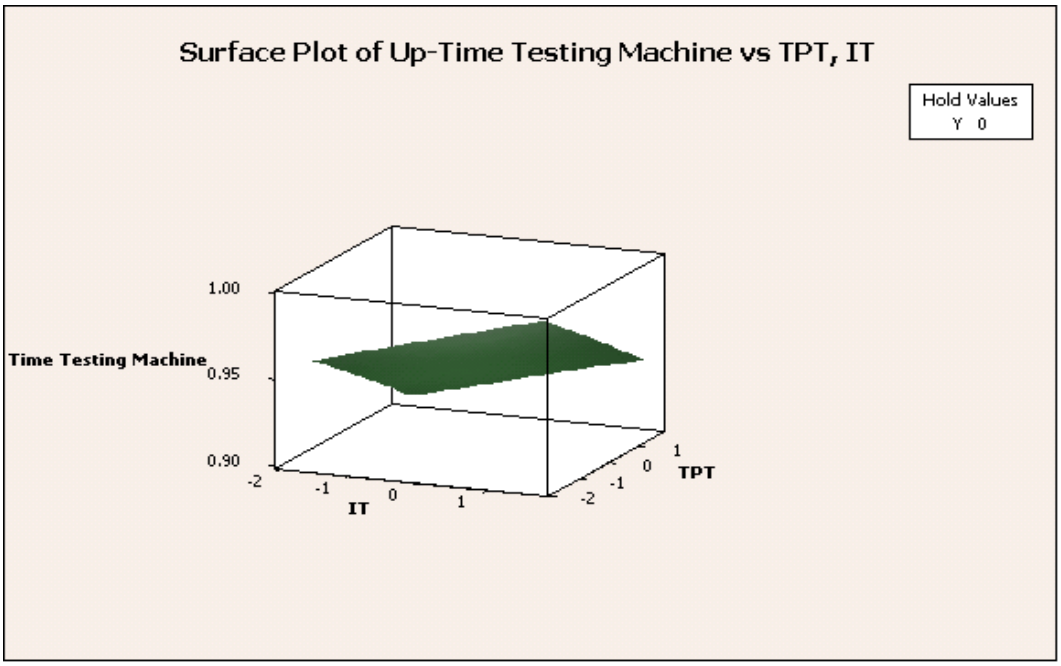

Figure 4.32 Surface plot for Uptime testing machine vs. inter-arrival time and testing process time – 3 testing machines.

4.4.1.3.Work-in-Process Inventory

Figure 4.33 is an output summary from Minitab for the coefficients of the regression and the analysis of variance. The analysis of the coefficients suggested that the model had a quadratic behavior. Note that the block element was also considered as critical; this is as a result of using new set of seeds (for the random generators) for each new replicate and having only two replicates at the cube points and axial points. The impact of the seeds could have been minimized with more replicates. Nevertheless, the coefficient of the block was not included in the metamodel Equation (4.3), thus going into the error of the throughput estimate.

$$WIP_{3TM} = 20.2662 + 3.3486 * IT + 0.2924 * Y - 4.4936 * TPT + \\ + 0.5642 * IT^2 + 1.2536 * TPT^2 - 1.8875 * IT * TPT \quad (4.3)$$

Where, WIP_{3TM} is the estimate for the work-in-process inventory when 3 testing machines are used, IT is the inter-arrival time, Y is the yield, and TPT is the testing process time.

```
The analysis was done using coded units.

Estimated Regression Coefficients for WIP

Term          Coef  SE Coef         T      P
Constant   20.2662   0.2055   98.621  0.000
Block      -0.4500   0.1127   -3.994  0.000
IT          3.3486   0.1363   24.560  0.000
Y           0.2924   0.1363    2.145  0.040
TPT        -4.4936   0.1363  -32.958  0.000
IT*IT       0.5642   0.1327    4.251  0.000
Y*Y         0.1134   0.1327    0.855  0.400
TPT*TPT     1.2536   0.1327    9.445  0.000
IT*Y        0.0125   0.1781    0.070  0.945
IT*TPT     -1.8875   0.1781  -10.596  0.000
Y*TPT      -0.1250   0.1781   -0.702  0.488

S = 0.7126   R-Sq = 98.5%   R-Sq(adj) = 98.0%


Analysis of Variance for WIP

Source           DF   Seq SS   Adj SS   Adj MS       F      P
Blocks            1    8.100    8.100    8.100   15.95  0.000
Regression        9  968.387  968.387  107.599  211.92  0.000
  Linear          3  860.129  860.129  286.710  564.68  0.000
  Square          3   51.003   51.003   17.001   33.48  0.000
  Interaction     3   57.255   57.255   19.085   37.59  0.000
Residual Error   29   14.724   14.724    0.508
  Lack-of-Fit    19    8.776    8.776    0.462    0.78  0.696
  Pure Error     10    5.948    5.948    0.595
Total            39  991.211
```

Figure 4.33 Regression analysis for WIP – 3 testing machines.

It was confirmed by the analysis of variance that the model had a quadratic behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the work-in-process inventory. The assumptions of normality and randomness of the residuals related to the analysis of variance were validated. Figure 4.34 presents several plots related to the normality and randomness of the residuals, such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals, residuals vs. the order of the data. Some concerns were raised with respect to the residuals vs. the fitted values since the spread of the data seemed to increase as the fitted value increased. This could meant that the standard deviations were not the same at each experimental design point, thus suggesting that the model was loosing accuracy as it tried to estimate higher values for the work-in-process. Nevertheless, no further investigation was requested based on the results of the lack of fit test and the R^2_{adj}, and the fact that model accuracy was not critical for estimating values of the work-in-process greater than 22 units, which was the constraint defined by the

management of the company. It was decided then to perform a normality test and runs test. Figure 4.35 and Figure 4.36 shows the results of those tests. The normality and randomness assumptions were validated (observe *p-values > 0.05*).

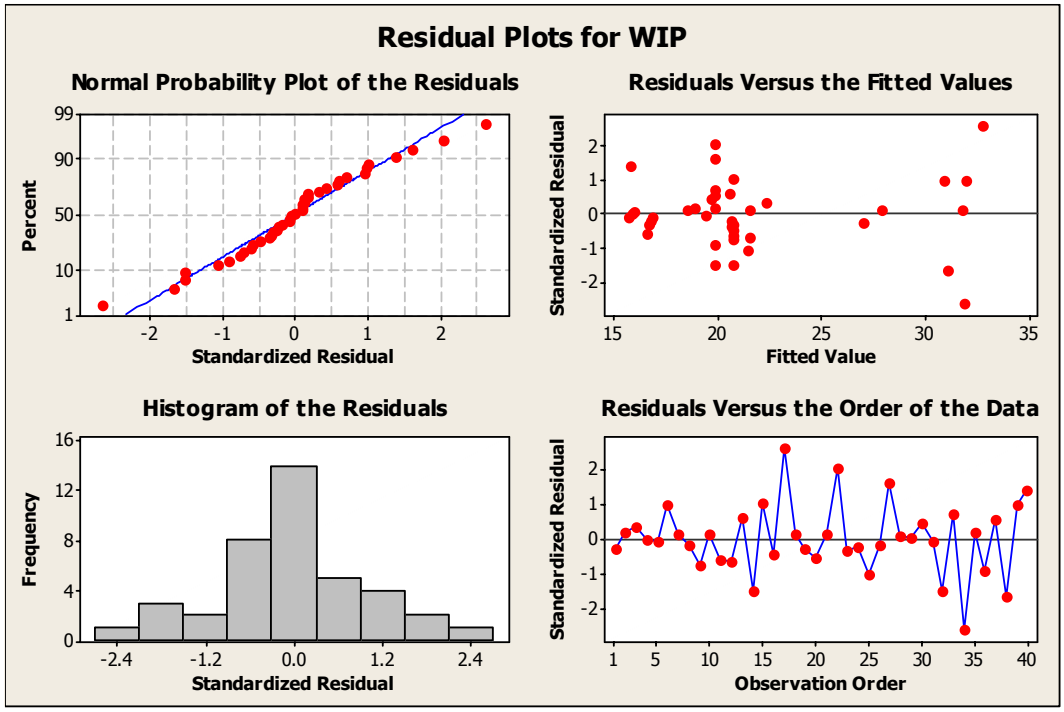

Figure 4.34 Residual plots for WIP – 3 testing machines.

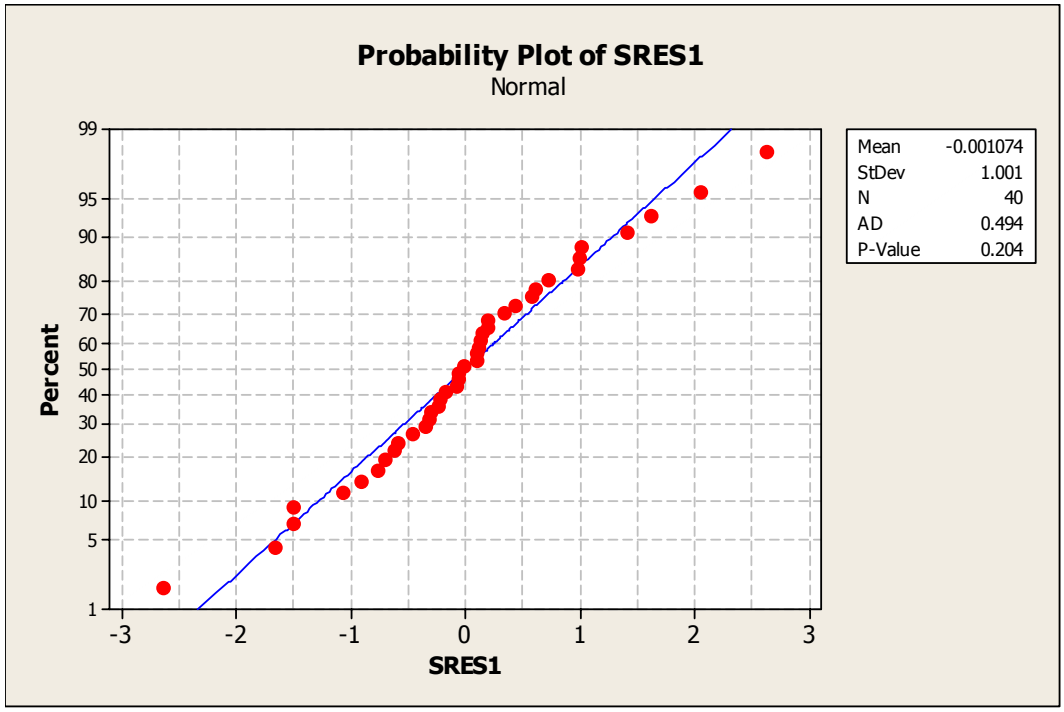

Figure 4.35 Normal probability plot for WIP – 3 testing machines.

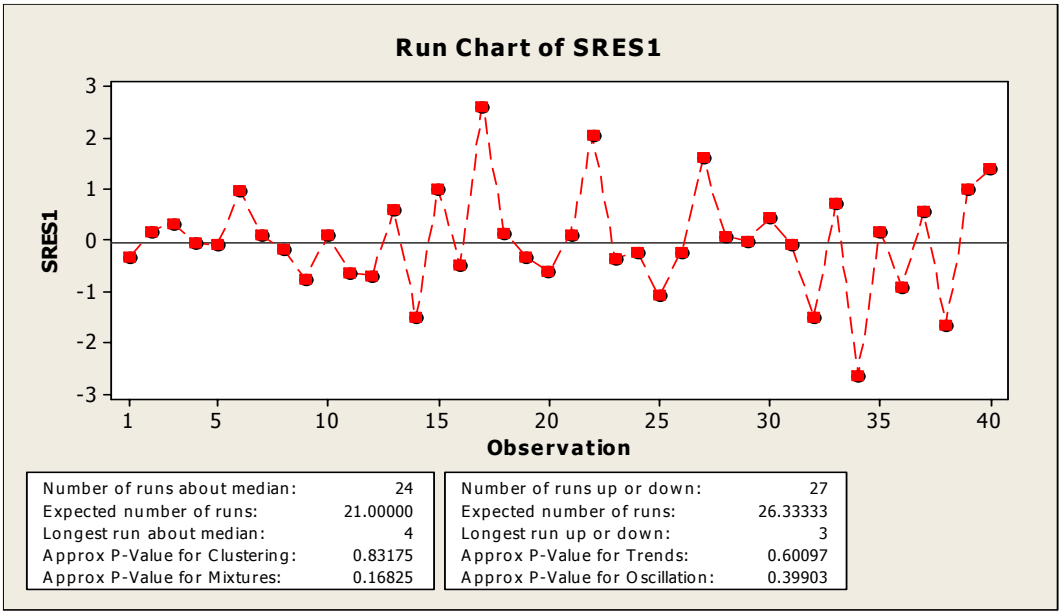

Figure 4.36 Runs test for WIP – 3 testing machines.

Figures 4.37, 4.38, and 4.39 are surface plots for the critical factors or input parameters (inter-arrival time, yield and testing process time); note the quadratic relationship with respect to the work-in-process inventory. It was concluded from Figure 4.37 that the testing process time was restricted at the lower level of the experimental region based on the work-in-process constraint of 22 units.

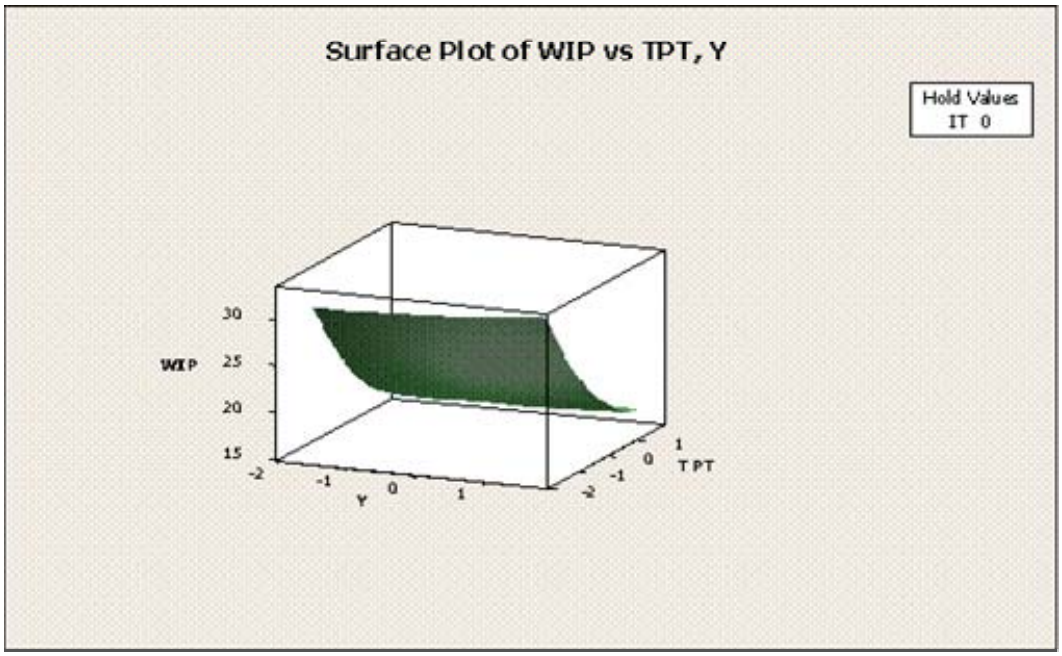

Figure 4.37 Surface plot for WIP vs. yield and testing process time – 3 testing machines.

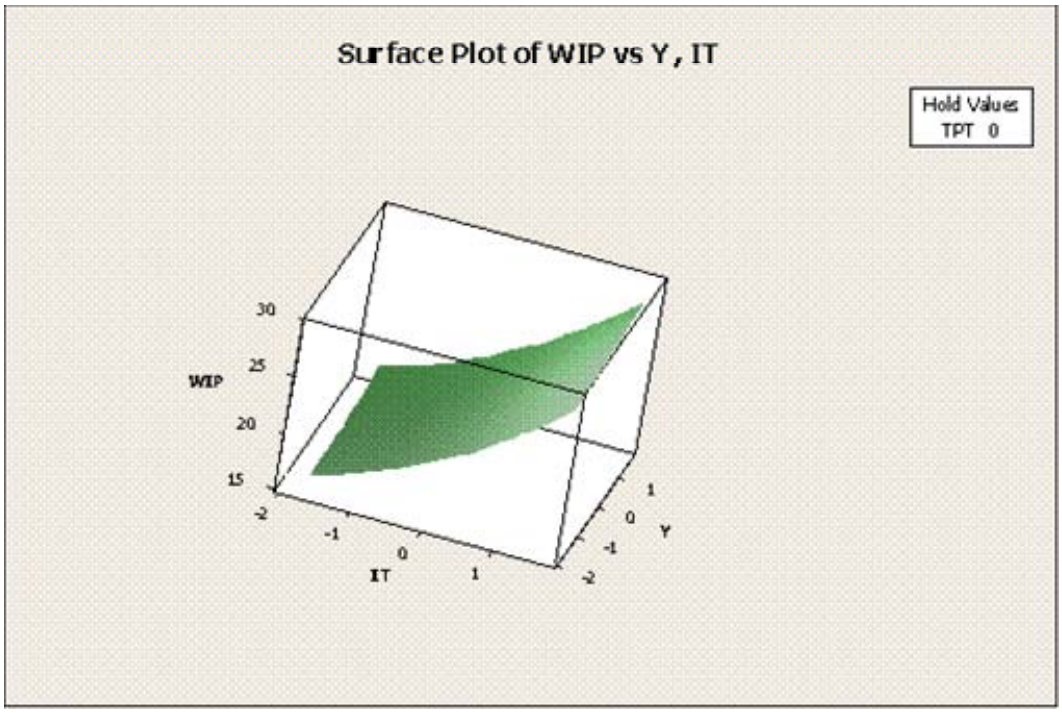

Figure 4.38 Surface plot for WIP vs. inter-arrival time and yield – 3 testing machines.

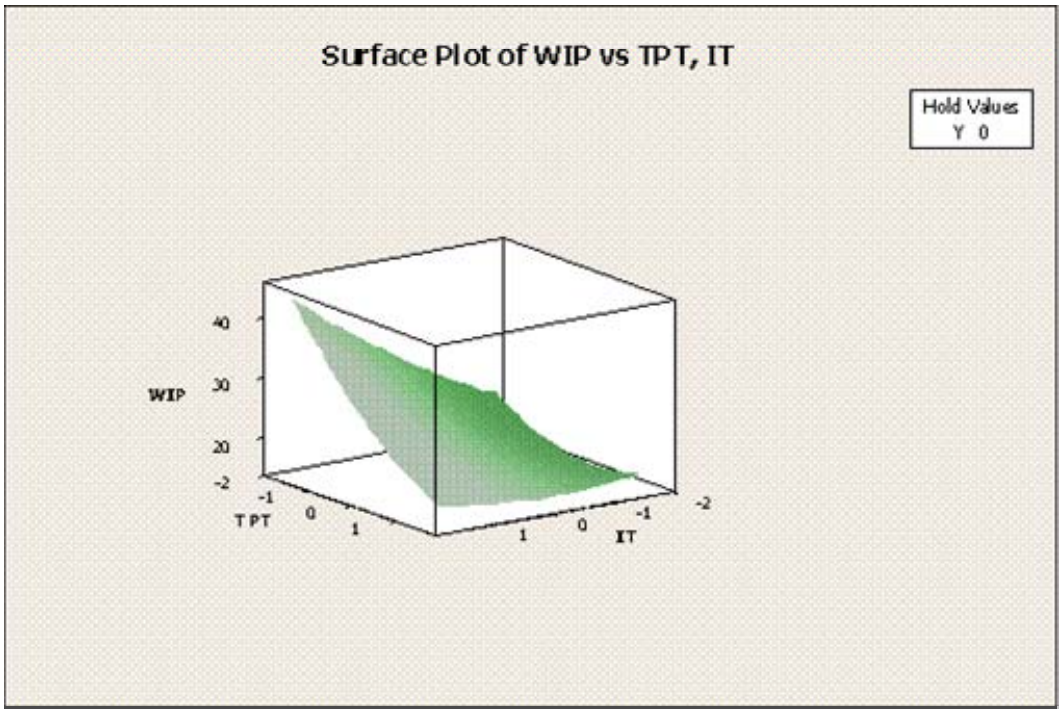

Figure 4.39 Surface plot for WIP vs. inter-arrival time and testing process time – 3 testing machines.

4.4.2. Metamodels for 4 Testing Machines

This section covers the results and conclusions related to the development of metamodels when 4 testing machines were used. As previously mentioned, the experiment was a central composite design with 3 factors and two levels at each factor, 2 replicates and 2 blocks.

The performance measures of interest were the throughput, the up-time for the testing machines, and the work-in-process inventory. The following analysis was done for each of these response variables: regression analysis, analysis of variance including the lack of fit test, normal probability plot for residuals, histogram of the residuals, plot of the residual vs. fitted values and plot of the residuals vs. the order of the data.

Table 4.25 presents the data obtained from the simulation output for each performance measures of interest. Note that the utilization of the assembly personnel, the final inspector and the packing personnel was never above the respective limits specified by the management of the company. In addition, the up-time for the draw out machine did not exceed the specified limit. For this reason, it was decided not to build metamodels for those variables.

After observing the throughput, minimum and average columns, it was decided to use the data of the average to build the metamodel. It was also decided to build metamodels for the up-time of the testing machines and the work-in-process inventory, since their respective limits were exceeded for some experimental runs.

Table 4.25 Data sheet for 2^3 central composite factorial design – 4 testing machines.

	Up-Time Testing Machine	Up-Time Draw Out	Utilization Assembly Personnel	Util. Final Inspector	Trough-put (Min.)	Through-put (Avg.)	Utilization Packing Personnel	WIP
Run Order	Model	Model	Model	Model	Model	Model	Model	Model
1	0.834	0.624	0.732	0.605	959	1029.2	0.786	12.9
2	0.777	0.665	0.762	0.666	1017	1089.3	0.832	13.0
3	0.894	0.633	0.744	0.614	972	1046.2	0.799	15.0
4	0.777	0.618	0.723	0.599	927	1016.5	0.776	11.9
5	0.742	0.635	0.736	0.625	984	1042.2	0.795	11.9
6	0.933	0.615	0.732	0.583	937	1015.7	0.776	17.9
7	0.868	0.650	0.755	0.641	996	1068.1	0.815	14.4
8	0.705	0.627	0.728	0.615	953	1029.9	0.786	11.4
9	0.832	0.622	0.729	0.603	933	1025.1	0.783	12.8
10	0.835	0.663	0.763	0.663	1005	1088.4	0.831	13.8
11	0.832	0.621	0.729	0.601	929	1022.7	0.781	12.7
12	0.892	0.589	0.707	0.546	911	975.8	0.745	14.1
13	0.834	0.587	0.699	0.549	904	969.6	0.740	12.1
14	0.832	0.623	0.730	0.604	922	1026.8	0.784	12.7
15	0.837	0.626	0.733	0.608	948	1032.5	0.788	12.9
16	0.834	0.622	0.729	0.602	942	1024.0	0.782	12.7
17	0.932	0.661	0.769	0.654	1012	1090.6	0.832	18.9
18	0.964	0.622	0.741	0.587	959	1028.4	0.785	24.1
19	0.804	0.603	0.713	0.576	939	994.6	0.759	11.9
20	0.742	0.592	0.698	0.563	903	976.7	0.746	10.9
21	0.771	0.615	0.719	0.595	918	1012.6	0.774	11.8
22	0.833	0.623	0.730	0.604	937	1028.0	0.785	12.7
23	0.892	0.586	0.705	0.541	900	970.8	0.741	14.2
24	0.864	0.647	0.751	0.638	979	1065.6	0.813	14.2
25	0.892	0.632	0.743	0.612	963	1042.8	0.795	15.0
26	0.833	0.661	0.761	0.659	1000	1083.6	0.827	13.7
27	0.835	0.625	0.733	0.605	964	1029.2	0.786	12.8
28	0.742	0.636	0.736	0.626	966	1043.7	0.796	11.9
29	0.704	0.626	0.725	0.611	945	1025.3	0.783	11.3
30	0.832	0.585	0.697	0.546	884	965.8	0.737	12.1
31	0.741	0.590	0.696	0.561	920	971.8	0.742	10.9
32	0.833	0.624	0.731	0.603	946	1026.9	0.783	12.8
33	0.840	0.629	0.736	0.612	962	1037.6	0.793	13.2
34	0.929	0.657	0.765	0.648	988	1083.1	0.827	18.2
35	0.772	0.662	0.758	0.661	987	1082.3	0.826	13.0
36	0.828	0.618	0.726	0.596	947	1017.5	0.777	12.6
37	0.839	0.627	0.734	0.611	950	1034.0	0.789	13.0
38	0.929	0.612	0.729	0.577	922	1013.0	0.774	17.3
39	0.960	0.619	0.739	0.582	935	1023.6	0.781	23.1
40	0.803	0.600	0.709	0.572	898	990.1	0.756	11.8

4.4.2.1.Throughput

Figure 4.40 is an output summary from Minitab for the coefficients of the regression and the analysis of variance.

Response Surface Regression: Throughput (Avg.) versus Block, IT, Y, TPT

The analysis was done using coded units.

Estimated Regression Coefficients for Throughput (Avg.)

Term	Coef	SE Coef	T	P
Constant	1027.73	1.1381	903.039	0.000
Block	-1.13	0.6240	-1.803	0.082
IT	21.37	0.7551	28.301	0.000
Y	35.34	0.7551	46.800	0.000
TPT	0.09	0.7551	0.120	0.905
IT*IT	1.02	0.7351	1.393	0.174
Y*Y	0.05	0.7351	0.070	0.945
TPT*TPT	0.03	0.7351	0.046	0.964
IT*Y	0.48	0.9866	0.488	0.629
IT*TPT	-0.03	0.9866	-0.032	0.975
Y*TPT	-0.47	0.9866	-0.475	0.638

S = 3.946 R-Sq = 99.0% R-Sq(adj) = 98.7%

Analysis of Variance for Throughput (Avg.)

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Blocks	1	50.6	50.6	50.6	3.25	0.082
Regression	9	46619.8	46619.8	5180.0	332.62	0.000
Linear	3	46582.2	46582.2	15527.4	997.05	0.000
Square	3	30.4	30.4	10.1	0.65	0.589
Interaction	3	7.2	7.2	2.4	0.15	0.926
Residual Error	29	451.6	451.6	15.6		
Lack-of-Fit	19	149.3	149.3	7.9	0.26	0.994
Pure Error	10	302.3	302.3	30.2		
Total	39	47122.0				

Figure 4.40 Regression analysis for throughput – 4 testing machines.

The analysis of the coefficients suggested that the model had a linear behavior since only main factors (inter-arrival time and yield) were identified as critical elements of the model. The metamodel was defined as Equation (4.4).

$$Throughput_{4TM} = 1027.73 + 21.37 * IT + 35.34 * Y \quad (4.4)$$

Where, $Throughput_{4TM}$ is the estimate for the throughput when 4 testing machines are used, IT is the inter-arrival time, and Y is the yield.

It was confirmed by the analysis of variance that the model had a linear behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or

metamodel was a good estimator of the throughput. Figure 4.41 presents several plots related to the normality and randomness of the residuals, such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals and residuals vs. the order of the data.

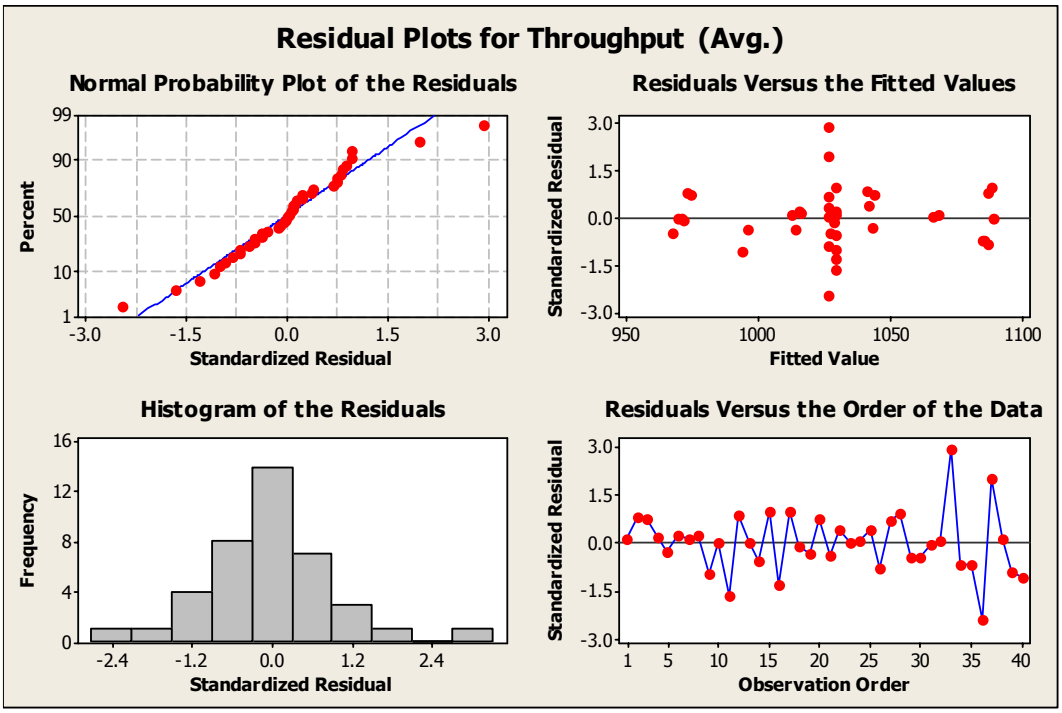

Figure 4.41 Residual plots for throughput – 4 testing machines.

The major concern with regards to the validation was the plot of the residuals vs. the fitted values. It was observed that the variance at the middle point was higher than the rest of the chart. It was concluded after detailed analysis of the data that this behavior was caused by the random number generators. It was observed that those values were representative of the center points in the experimental cube; by design, the number of runs in the center point was 12 while there were only 2 replicates at the cube points and at the axial points. Since new seeds were being used for every new replicate, ten more seeds were used in the center point, thus introducing more variability as compared to the cube points and axial points. Table 4.26 shows the throughput obtained from the simulation run for each replicate at the center point. Two important things were observed: (1) all values were above the constraint of 1000 units, and (2) the difference between the minimum and the maximum values was 17 units. It was concluded based on this

information, that the risk of using this data for the construction of the metamodel was low. A difference of 17 units was considered negligible as compared to 1000 units, and the probability of rejecting an alternative based on the throughput constraint was also considered to be low since all values obtained from the simulation were above a 1000 units.

Table 4.26 Data sheet for center points 2^3 central composite factorial design – 4 testing machines

	Experimental Factors/Simulation Parameters			Up-Time Testing Machine	Throughput (Avg.)	WIP
Run Order	IT	Y	TPT	Simulation Model	Simulation Model	Simulation Model
1	0.0000	0.0000	0.0000	0.834	1029.2	12.9
9	0.0000	0.0000	0.0000	0.832	1025.1	12.8
11	0.0000	0.0000	0.0000	0.832	1022.7	12.7
14	0.0000	0.0000	0.0000	0.832	1026.8	12.7
15	0.0000	0.0000	0.0000	0.837	1032.5	12.9
16	0.0000	0.0000	0.0000	0.834	1024.0	12.7
22	0.0000	0.0000	0.0000	0.833	1028.0	12.7
27	0.0000	0.0000	0.0000	0.835	1029.2	12.8
32	0.0000	0.0000	0.0000	0.833	1026.9	12.8
33	0.0000	0.0000	0.0000	0.840	1037.6	13.2
36	0.0000	0.0000	0.0000	0.828	1017.5	12.6
37	0.0000	0.0000	0.0000	0.839	1034.0	13.0

Normality test and runs test were done in order to gain more confidence with respect to the validation of the assumptions from the analysis of variance. Figure 4.42 and Figure 4.43 shows the results of those tests. It was assumed that both assumptions were validated based on the results of these tests (observe $p > 0.05$) and the explanation provided in the previous paragraph.

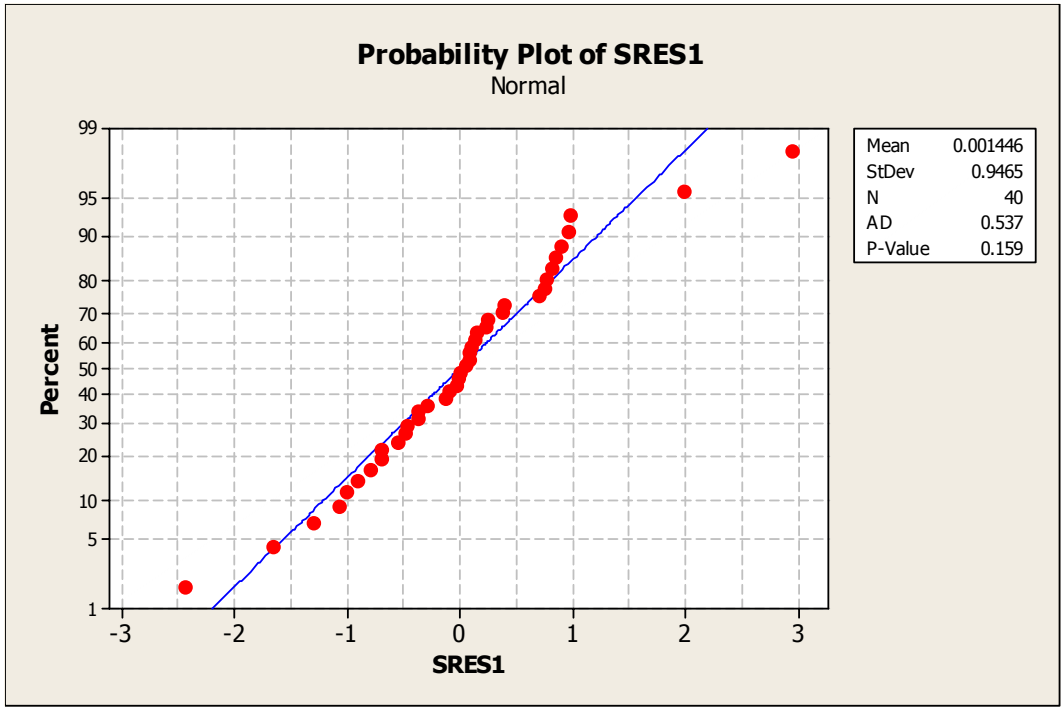

Figure 4.42 Normal probability plot for throughput – 4 testing machines

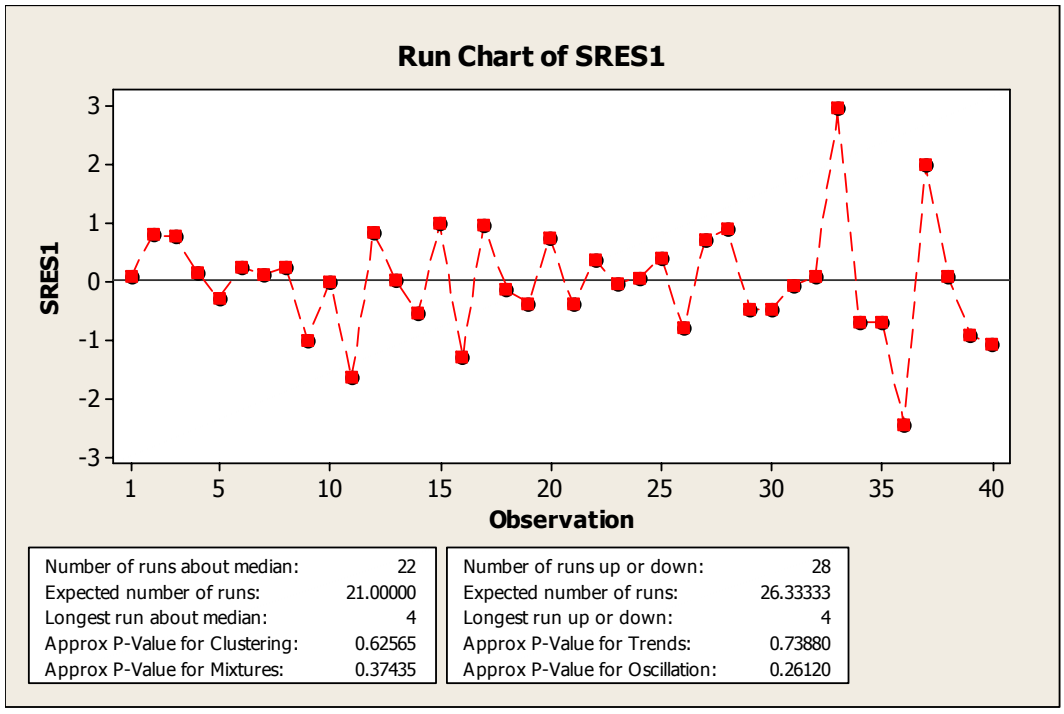

Figure 4.43 Runs test for throughput – 4 testing machines.

Figure 4.44 is a surface plot for the critical factors (inter-arrival time and yield); note the linear relation with respect to the throughput. It was observed that the throughput increases as both factors moved toward their respective high levels of the experimental region. Note that the yield was restricted to its high level by definition of the constraint specified by the management of the company.

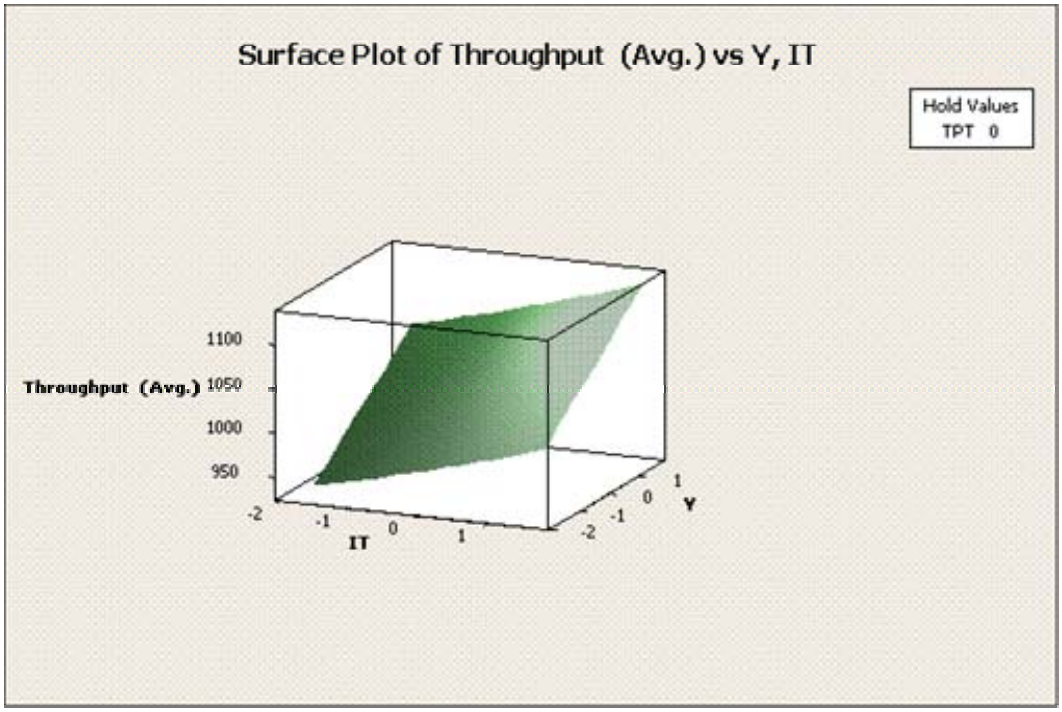

Figure 4.44 Surface plot for throughput vs. inter-arrival time and yield – 4 testing machines.

4.4.2.2.Up-time Testing Machines

Figure 4.45 is an output summary from Minitab for the coefficients of the regression and the analysis of variance. The analysis of the coefficients suggested that the model had a linear behavior since only main factors (inter-arrival time and testing process time) were identified as critical elements of the model with their respective interaction. Equation (4.5) is the metamodel for the up-time of the testing machines.

$$Up-timeTestingMachines_{4TM} = 0.834051 + 0.018058 * IT - 0.076706 * TPT - 0.001437 * IT * TPT \quad (4.5)$$

Where, $Up\text{-}timeTesting\ Machines_{4TM}$ is the estimate for the testing machines up-time when 4 testing machines are used, *IT* is the inter-arrival time, and *TPT* is the testing process time.

Response Surface Regression: Up-Time Testing Machine versus Block, IT, Y, TPT

The analysis was done using coded units.

Estimated Regression Coefficients for Up-Time Testing Machine

Term	Coef	SE Coef	T	P
Constant	0.834051	0.000720	1159.203	0.000
Block	-0.000700	0.000394	-1.775	0.086
IT	0.018058	0.000477	37.827	0.000
Y	0.000233	0.000477	0.488	0.629
TPT	-0.076706	0.000477	-160.683	0.000
IT*IT	0.000446	0.000465	0.959	0.346
Y*Y	0.000004	0.000465	0.008	0.994
TPT*TPT	-0.000085	0.000465	-0.182	0.856
IT*Y	-0.000187	0.000624	-0.301	0.766
IT*TPT	-0.001437	0.000624	-2.305	0.029
Y*TPT	0.000062	0.000624	0.100	0.921

S = 0.002495 R-Sq = 99.9% R-Sq(adj) = 99.9%

Analysis of Variance for Up-Time Testing Machine

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Blocks	1	0.000020	0.000020	0.000020	3.15	0.086
Regression	9	0.169656	0.169656	0.018851	3028.51	0.000
Linear	3	0.169616	0.169616	0.056539	9083.38	0.000
Square	3	0.000006	0.000006	0.000002	0.33	0.801
Interaction	3	0.000034	0.000034	0.000011	1.80	0.169
Residual Error	29	0.000181	0.000181	0.000006		
Lack-of-Fit	19	0.000064	0.000064	0.000003	0.29	0.991
Pure Error	10	0.000117	0.000117	0.000012		
Total	39	0.169856				

Figure 4.45 Regression analysis for Uptime testing machine – 4 testing machines.

It was confirmed by the analysis of variance that the model had a linear behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the up-time for the testing machines. Figure 4.46 was built with the purpose of validating the assumptions of the analysis of variance related to the normality and randomness of the residuals. This figure presents several plots such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals and residuals vs. the order of the data. The major concern with regards to the validation was the plot of the residuals vs. the fitted values. It was observed that the variance at the middle point was higher than the rest of the chart. It was concluded after detailed analysis of the data that this behavior was caused by the random number generators. It was

observed that those values were representative of the center points in the experimental cube; by design, the number of runs in the center point was 12 while there were only 2 replicates at the cube points and at the axial points. Since new seeds were being used for every new replicate, ten more seeds were used in the center point, thus introducing more variability as compared to the cube points and axial points. Table 4.26 shows the testing machines up-time obtained from the simulation run for each replicate at the center point. Two important things were observed: (1) all values were below the constraint of 95%, and (2) the difference between the minimum (82.8%) and the maximum (84%) values was 1.2%. It was concluded based on this information, that the risk of using this data for the construction of the metamodel was low. A difference of 1.2% was considered negligible as compared to 95% when the data was between the range of 82.8% and 84.0%. It was concluded that the probability of rejecting an alternative based on the up-time constraint was low.

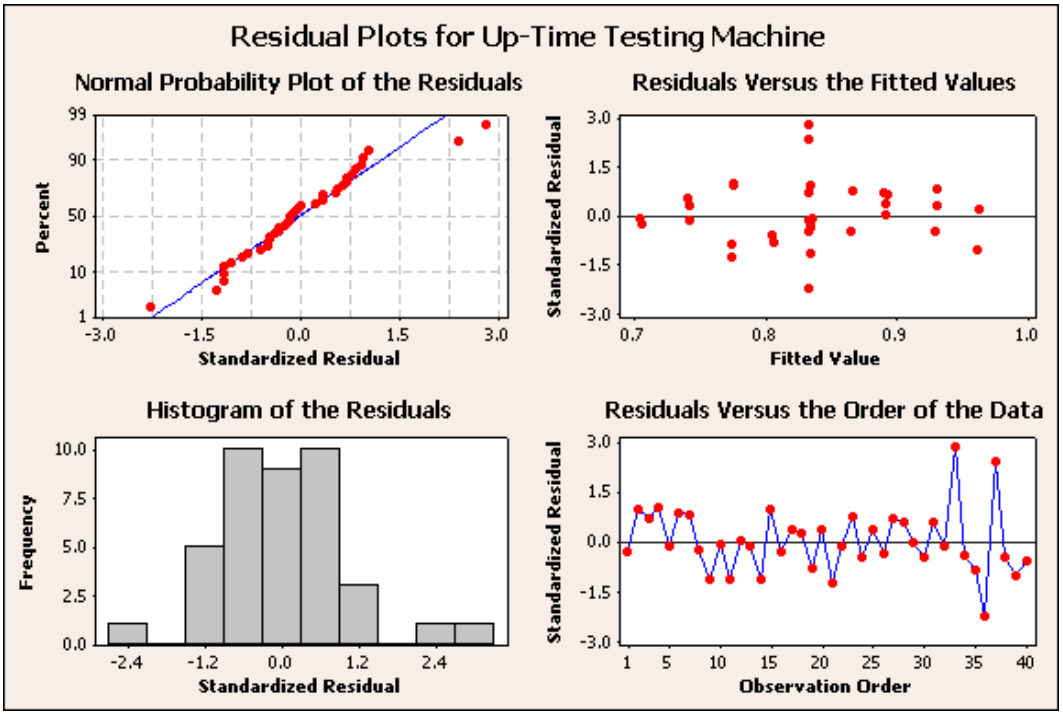

Figure 4.46 Residual plots for Uptime testing machine – 4 testing machines.

In addition to the explanation offered in the previous paragraph with respect to the assumptions of the analysis of variance, normality test and a runs test were done. Figure 4.47 and Figure 4.48 shows the results of those tests. It was concluded that the

assumption of normality and randomness of the residuals were validated (observe $p > 0.05$).

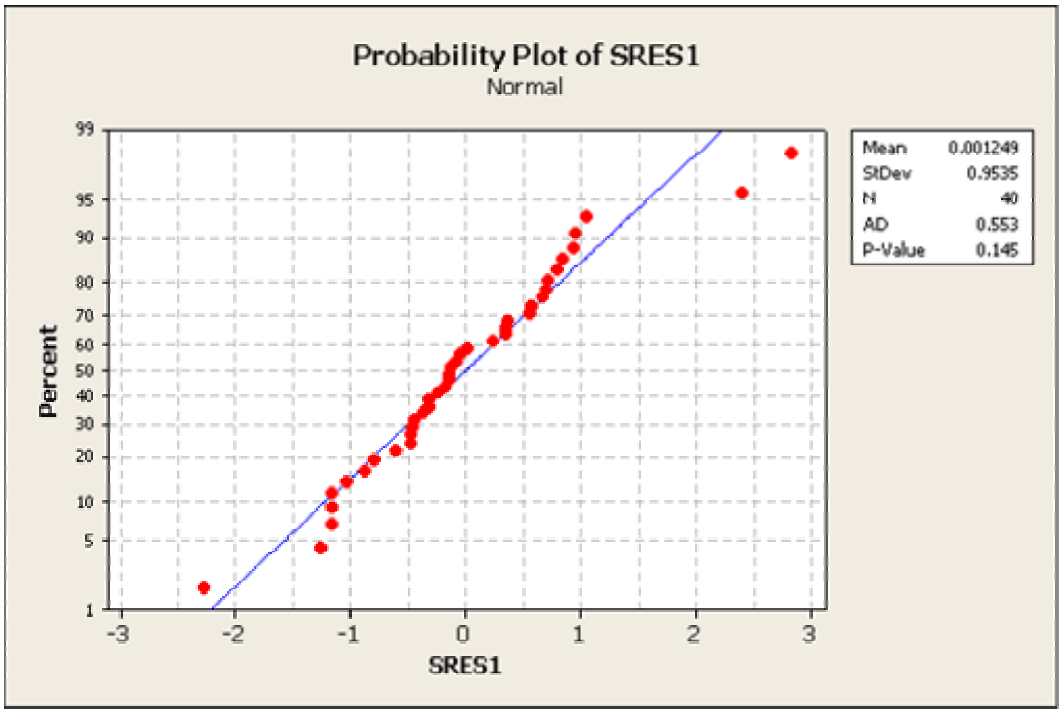

Figure 4.47 Normal probability plot for Uptime testing machine – 4 testing machines.

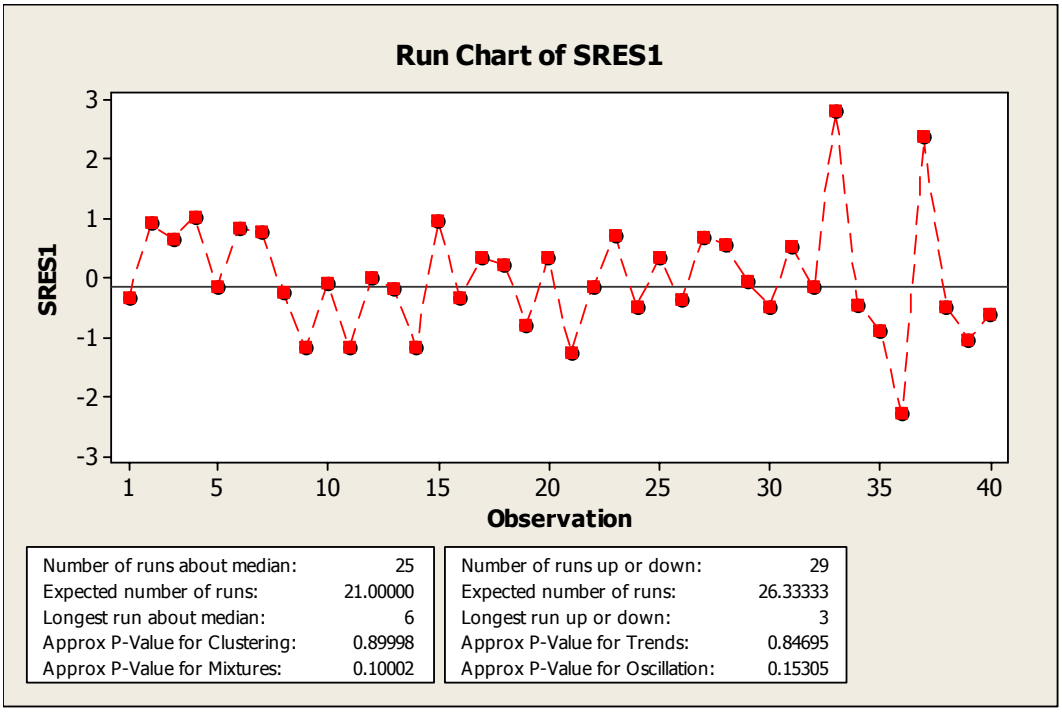

Figure 4.48 Runs test for Uptime testing machine – 4 testing machines.

Figure 4.49 is a surface plot for the critical factors (inter-arrival time and testing process time); note the linear relation with respect to the up-time of the testing machines. Note that the up-time for the testing machines is 95% as the testing process time factor approaches its lower level on the experimental region. Thus, it was concluded that the testing process time factor was restricted at both sides of the experimental region since the high level was the constraint specified by the management.

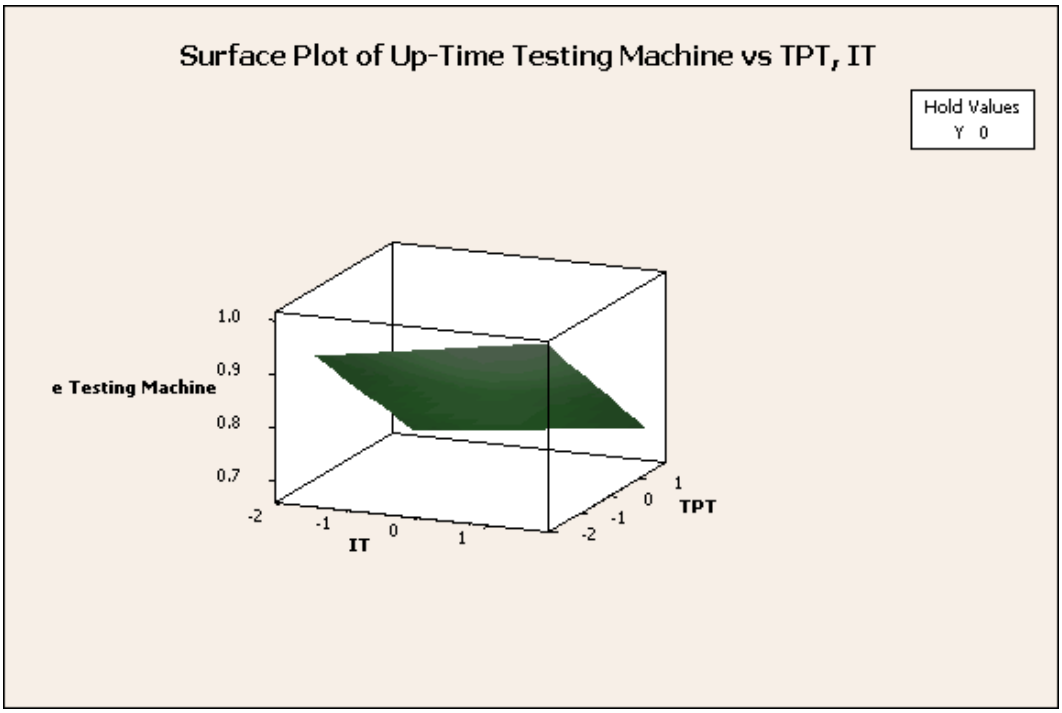

Figure 4.49 Surface plot for Uptime testing machine vs. inter-arrival time and testing process time – 4 testing machines.

4.4.2.3.Work-in-Process Inventory

Figure 4.50 is an output summary from Minitab for the coefficients of the regression and the analysis of variance. The analysis of the coefficients suggested that the model had a quadratic behavior but a major concerned was raised with the lack of fit test from the analysis of variance. This suggested that the model was not a good estimator for the work-in-process inventory. In addition, Figure 4.51 and Figure 4.52 showed that the assumptions of the analysis of variance with respect to the normality and the randomness of the residuals were not met. It was decided then to apply the Box-Cox transformation to the work-in-process data obtained from the simulation.

Response Surface Regression: WIP versus Block, IT, Y, TPT

The analysis was done using coded units.

Estimated Regression Coefficients for WIP

Term	Coef	SE Coef	T	P
Constant	12.8320	0.2227	57.612	0.000
Block	-0.0600	0.1221	-0.491	0.627
IT	0.9644	0.1478	6.526	0.000
Y	0.4924	0.1478	3.332	0.002
TPT	-2.8009	0.1478	-18.954	0.000
IT*IT	-0.0088	0.1439	-0.061	0.952
Y*Y	-0.0618	0.1439	-0.430	0.670
TPT*TPT	1.5468	0.1439	10.753	0.000
IT*Y	0.0312	0.1931	0.162	0.873
IT*TPT	-0.6187	0.1931	-3.205	0.003
Y*TPT	0.0437	0.1931	0.227	0.822

S = 0.7723 R-Sq = 94.9% R-Sq(adj) = 93.2%

Analysis of Variance for WIP

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Blocks	1	0.144	0.144	0.1440	0.24	0.627
Regression	9	323.394	323.394	35.9327	60.24	0.000
Linear	3	246.308	246.308	82.1027	137.65	0.000
Square	3	70.914	70.914	23.6381	39.63	0.000
Interaction	3	6.172	6.172	2.0573	3.45	0.029
Residual Error	29	17.298	17.298	0.5965		
Lack-of-Fit	19	17.015	17.015	0.8955	31.61	0.000
Pure Error	10	0.283	0.283	0.0283		
Total	39	340.836				

Figure 4.50 Regression analysis for WIP – 4 testing machines

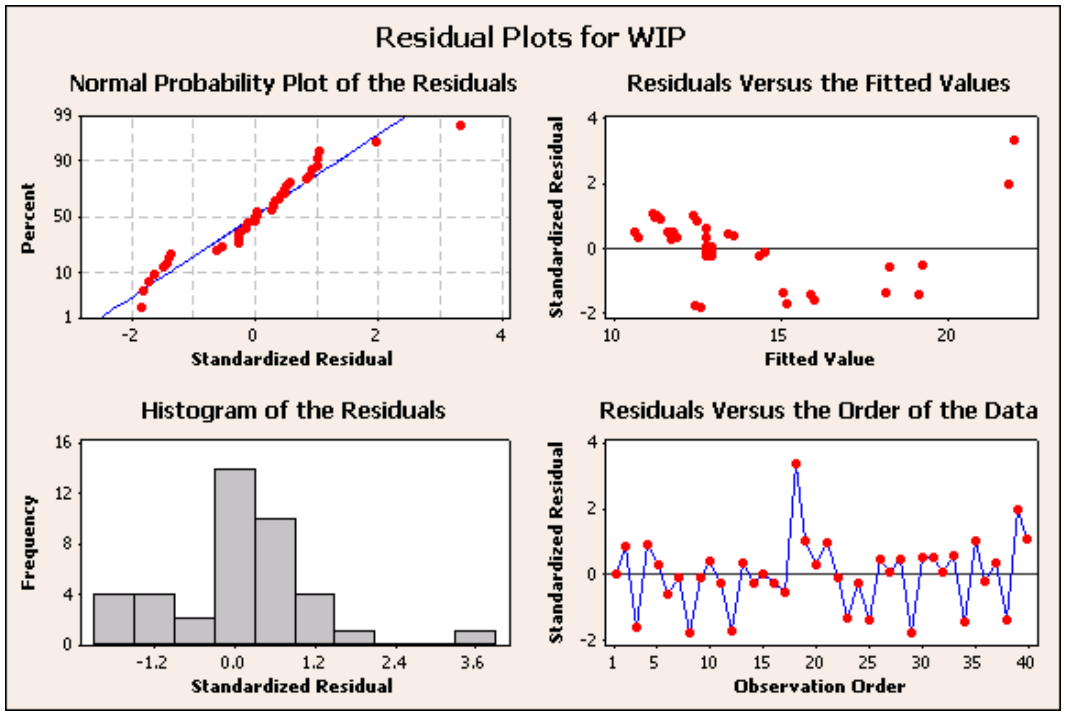

Figure 4.51 Residual plots for WIP – 4 testing machines.

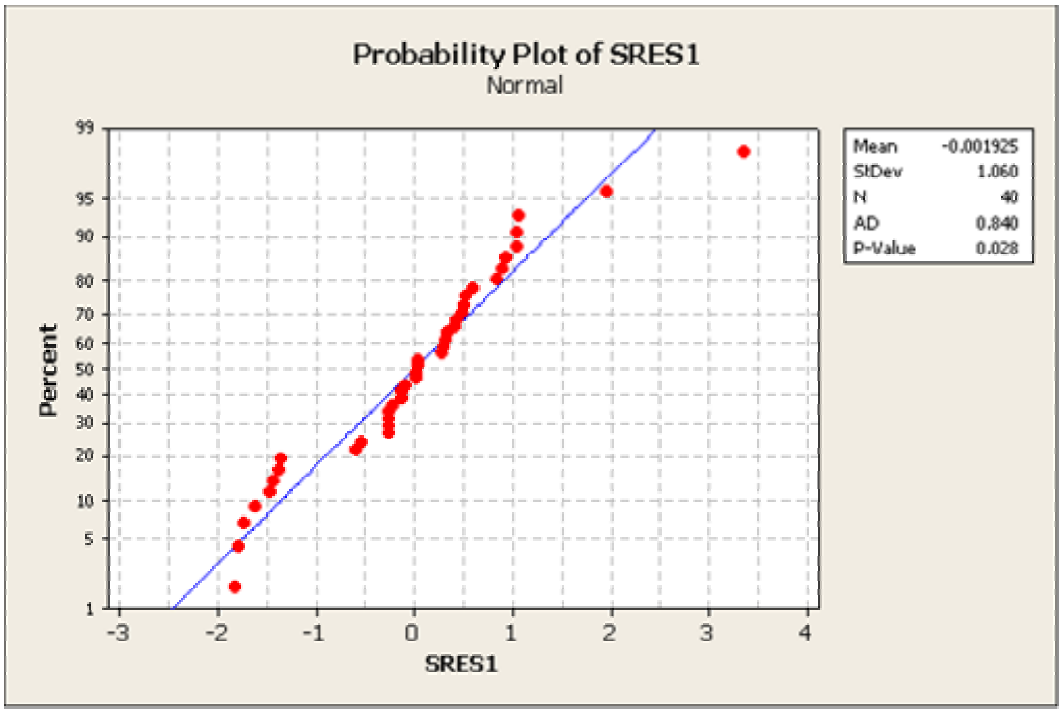

Figure 4.52 Normal probability plot for WIP – 4 testing machines.

Figure 4.53 is an output summary from Minitab for the transformed data.

Response Surface Regression: WIP- Transformed versus Block, IT, Y, TPT

The analysis was done using coded units.

Estimated Regression Coefficients for WIP- Transformed

Term	Coef	SE Coef	T	P
Constant	0.000256	0.000002	109.546	0.000
Block	0.000001	0.000001	0.742	0.464
IT	-0.000044	0.000002	-28.637	0.000
Y	-0.000030	0.000002	-19.577	0.000
TPT	0.000103	0.000002	66.156	0.000
IT*IT	-0.000002	0.000002	-1.488	0.147
Y*Y	-0.000002	0.000002	-1.179	0.248
TPT*TPT	-0.000019	0.000002	-12.536	0.000
IT*Y	0.000005	0.000002	2.416	0.022
IT*TPT	-0.000002	0.000002	-0.835	0.411
Y*TPT	-0.000018	0.000002	-9.026	0.000

S = 0.000008100 R-Sq = 99.5% R-Sq(adj) = 99.3%

Analysis of Variance for WIP- Transformed

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Blocks	1	0.000000	0.000000	0.000000	0.55	0.464
Regression	9	0.000000	0.000000	0.000000	647.25	0.000
Linear	3	0.000000	0.000000	0.000000	1860.01	0.000
Square	3	0.000000	0.000000	0.000000	52.41	0.000
Interaction	3	0.000000	0.000000	0.000000	29.34	0.000
Residual Error	29	0.000000	0.000000	0.000000		
Lack-of-Fit	19	0.000000	0.000000	0.000000	0.36	0.974
Pure Error	10	0.000000	0.000000	0.000000		
Total	39	0.000000				

Figure 4.53 Regression analysis for WIP – 4 testing machines.

The analysis of the coefficients suggested that the model had a quadratic behavior. The metamodel developed is presented in Equation (4.6).

$$WIP_{4TM} = \left[\frac{1}{\begin{pmatrix} 0.000256 - 0.000044 * IT - 0.00003 * Y + 0.000103 * TPT - \\ 0.000019 * TPT^2 + 0.000005 * IT * Y - 0.000018 * Y * TPT \end{pmatrix}} \right]^{\left(\frac{1}{3.24}\right)} \quad (4.6)$$

Where, WIP_{4TM} is the estimate for the work-in-process inventory when 4 testing machines are used, IT is the inter-arrival time, y is the yield, and TPT is the testing process time.

It was confirmed by the analysis of variance that the model had a quadratic behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the work-in-process inventory. Figure 4.54 presents several plots related to the normality and randomness of the residuals, such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals and residuals vs. the order of the data. The presence of an outlier was observed for the plot of residuals vs. the fitted values and the plot of residuals vs. the order of the data. After reviewing the data, it was decided to leave the datum as part of the analysis since it was found that it was a valid simulation output. A normality test and runs test were then performed. Figure 4.55 and Figure 4.56 shows the results of those tests. The normality and randomness assumptions were validated (observe *p-values > 0.05*).

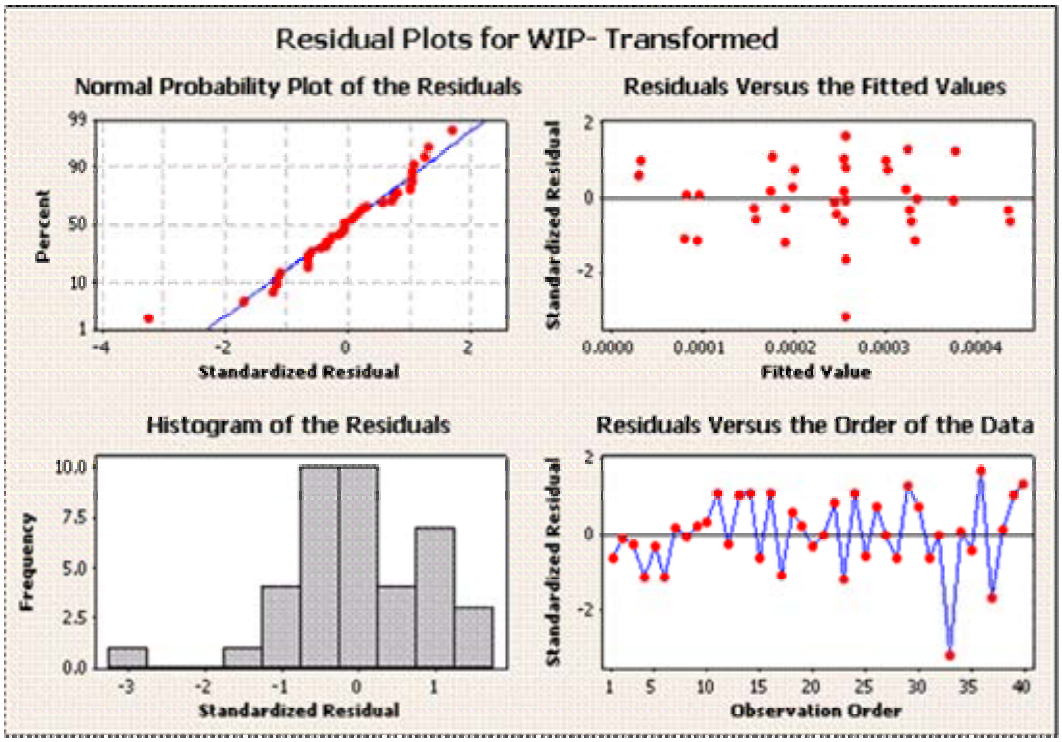

Figure 4.54 Residual plots for WIP – 4 testing machines

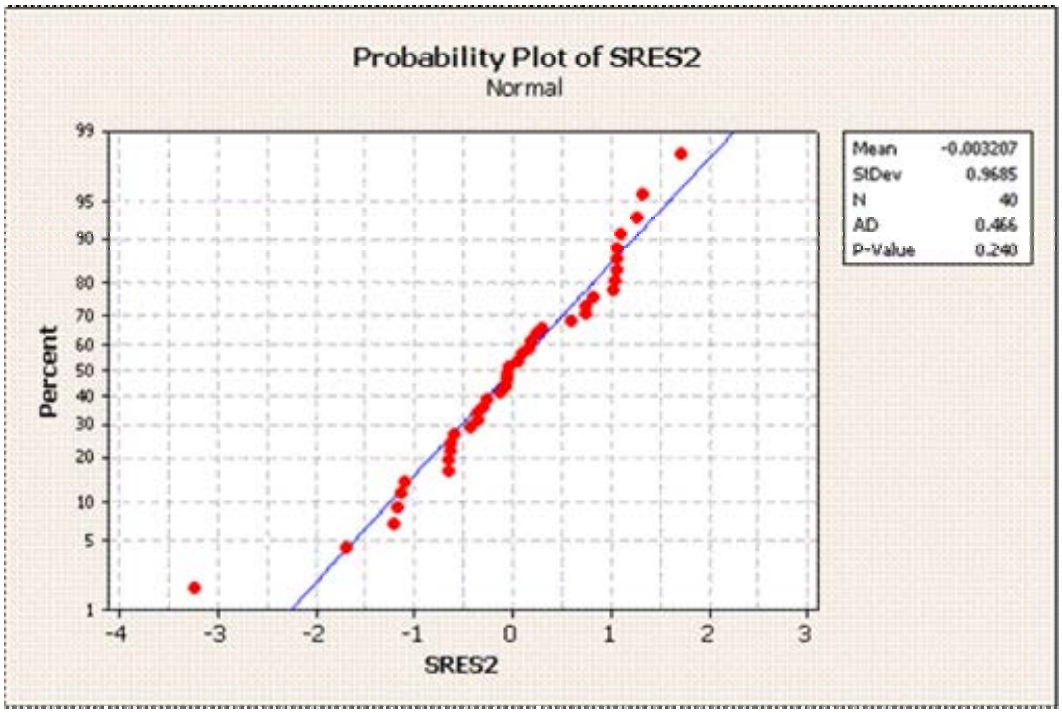

Figure 4.55 Normal probability plot for WIP – 4 testing machines.

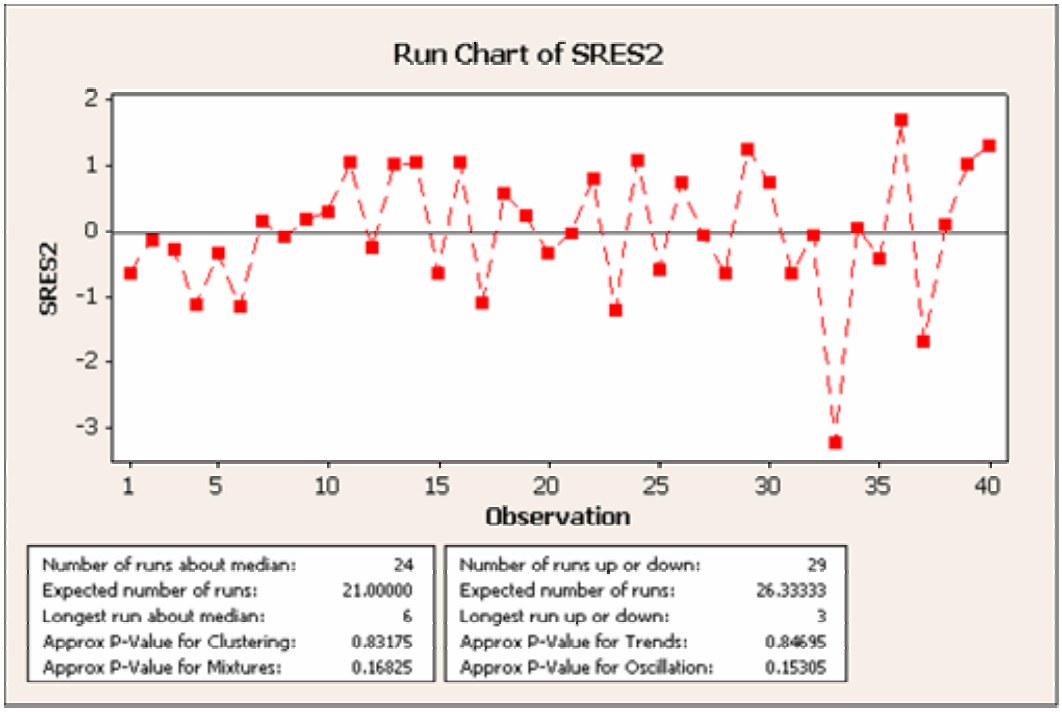

Figure 4.56 Runs test for WIP – 4 testing machines.

Figures 4.57, 4.58, and 4.59 are surface plots for the critical factors or input parameters (inter-arrival time, yield and testing process time); note the quadratic relationship with respect to the work-in-process inventory. Figure 4.57 validated the previous conclusion with respect to the testing process time being restricted within the experimental region; in other words, it can not expand outside the experimental region due to the work-in- process constraint of 22 units imposed by the management of the company. Figure 4.60 showed that the inter-arrival time was restricted by its higher level of the experimental region in order to meet the work-in-process constraint.

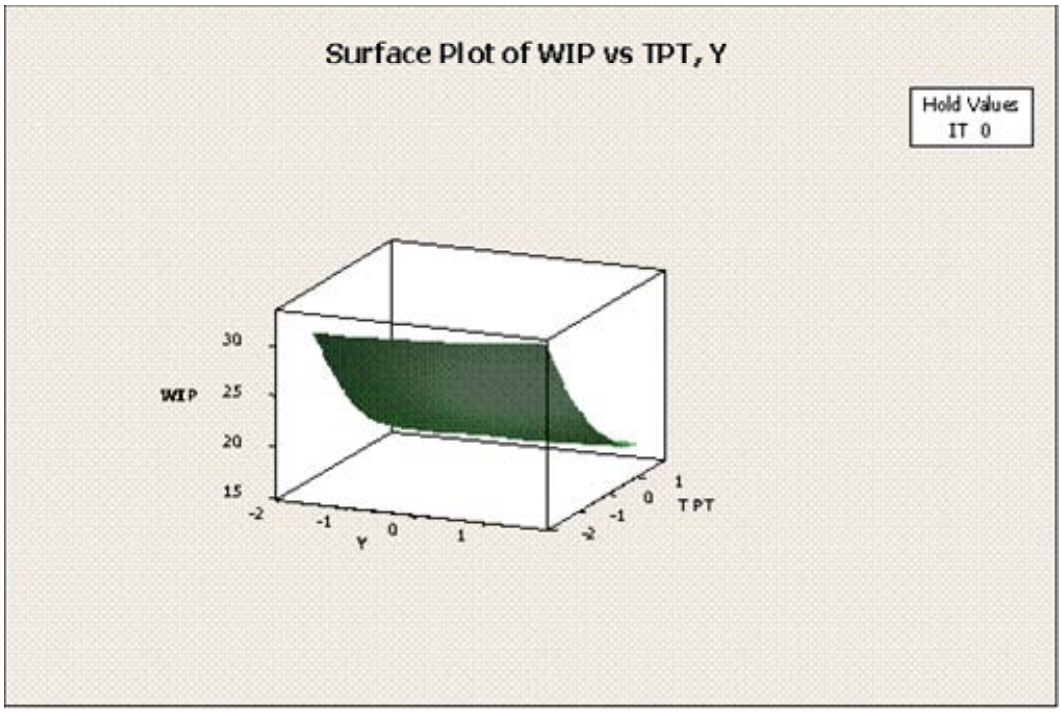

Figure 4.57 Surface plot for WIP vs. yield and testing process time – 4 testing machines.

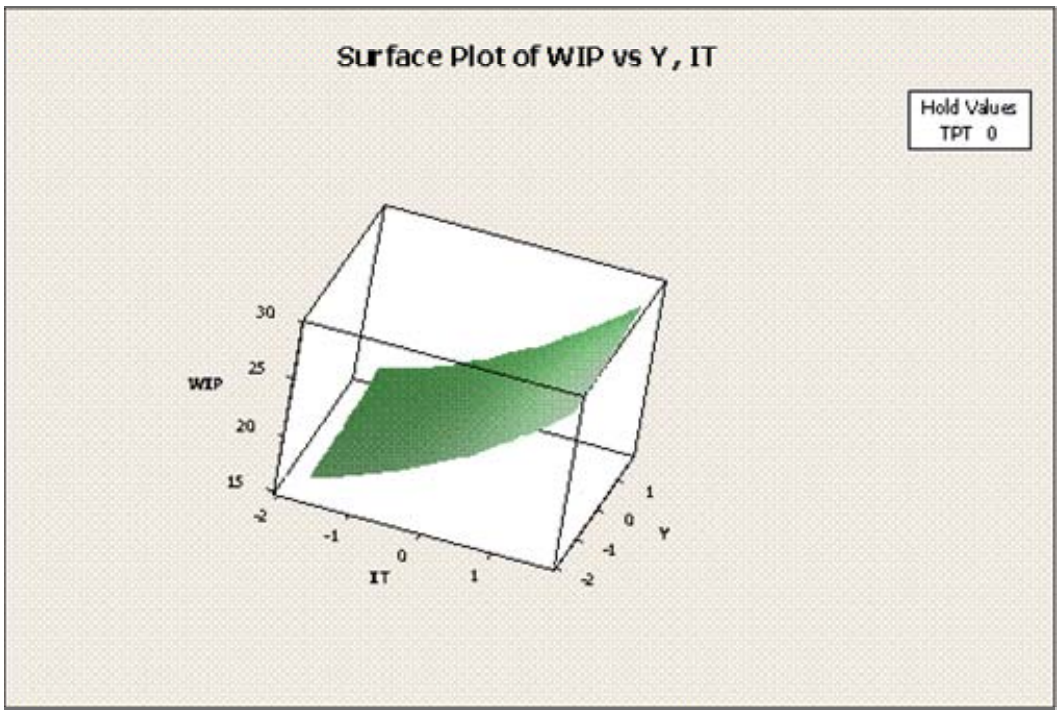

Figure 4.58 Surface plot for WIP vs. inter-arrival time and yield – 4 testing machines

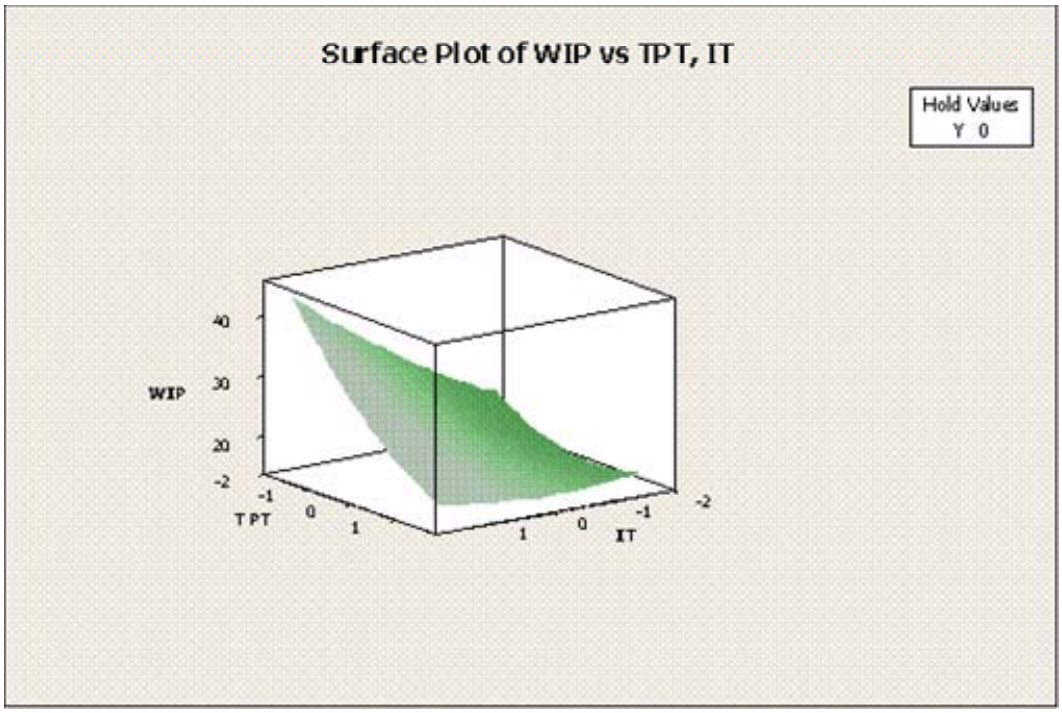

Figure 4.59 Surface plot for WIP vs. inter-arrival time and testing process time – 4 testing machines.

4.4.3. Metamodels for 5 Testing Machines

This section covers the results and conclusions related to the development of metamodels when 5 testing machines were used. As previously mentioned, the experiment was a central composite design with 3 factors and two levels at each factor, 2 replicates and 2 blocks.

The performance measures of interest were the throughput, the utilization of the packing personnel, and the work-in-process inventory. The following analysis was done for each of these response variables: regression analysis, analysis of variance including the lack of fit test, normal probability plot for residuals, histogram of the residuals, plot of the residual vs. fitted values, plot of the residuals vs. the order of the data.

Table 4.27 presents the data obtained from the simulation output for each performance measures of interest. Note that the utilization of the assembly personnel and the final inspector was never above the respective limits specified by the management of the company.

Table 4.27 Data sheet for 2^3 central composite factorial design – 5 testing machines.

	Up-Time Testing Machine	Up-Time Draw Out	Utilization Assembly Personnel	Utilization Final Inspector	Through-put (Min.)	Through-put (Avg.)	Utilization Packing Personnel	WIP
Run Order	Model	Model	Model	Model	Model	Model	Model	Model
1	0.704	0.632	0.730	0.622	946	1036.1	0.790	12.2
2	0.675	0.719	0.802	0.735	1076	1163.5	0.888	15.1
3	0.727	0.636	0.734	0.626	960	1040.6	0.794	12.5
4	0.675	0.625	0.723	0.611	928	1022.4	0.780	11.8
5	0.605	0.638	0.733	0.630	971	1041.6	0.795	11.6
6	0.812	0.622	0.726	0.605	901	1022.1	0.780	13.3
7	0.773	0.700	0.788	0.711	1071	1137.5	0.869	15.1
8	0.594	0.634	0.729	0.626	953	1035.3	0.790	11.5
9	0.702	0.631	0.729	0.619	952	1033.7	0.790	12.2
10	0.704	0.706	0.792	0.720	1054	1147.0	0.876	14.7
11	0.700	0.627	0.726	0.616	942	1028.7	0.786	12.0
12	0.727	0.557	0.666	0.517	837	919.7	0.702	10.9
13	0.703	0.561	0.667	0.523	854	925.2	0.706	10.7
14	0.700	0.630	0.728	0.619	960	1033.6	0.789	12.1
15	0.704	0.632	0.730	0.620	950	1035.1	0.790	12.2
16	0.702	0.629	0.727	0.617	956	1031.1	0.787	12.1
17	0.811	0.717	0.802	0.735	1096	1162.3	0.887	16.2
18	0.812	0.630	0.733	0.618	953	1036.4	0.791	13.4
19	0.645	0.578	0.681	0.548	881	951.6	0.727	10.5
20	0.604	0.556	0.661	0.521	830	917.7	0.700	9.9
21	0.673	0.624	0.721	0.610	950	1020.6	0.779	11.8
22	0.707	0.638	0.734	0.628	952	1042.7	0.795	12.3
23	0.725	0.554	0.662	0.513	826	914.6	0.698	10.8
24	0.772	0.700	0.788	0.710	1030	1136.5	0.868	15.2
25	0.725	0.634	0.733	0.624	956	1039.5	0.793	12.4
26	0.702	0.705	0.791	0.717	1033	1143.7	0.873	14.6
27	0.701	0.630	0.729	0.619	948	1032.9	0.789	12.1
28	0.603	0.636	0.731	0.626	951	1038.6	0.792	11.5
29	0.593	0.632	0.727	0.621	949	1033.4	0.789	11.4
30	0.702	0.559	0.665	0.521	848	921.5	0.703	10.7
31	0.602	0.554	0.659	0.519	822	913.3	0.697	9.9
32	0.702	0.631	0.730	0.620	962	1034.3	0.790	12.1
33	0.701	0.631	0.729	0.620	964	1032.6	0.787	12.1
34	0.810	0.716	0.802	0.733	1074	1161.2	0.886	16.2
35	0.674	0.718	0.801	0.733	1071	1160.9	0.886	15.1
36	0.701	0.629	0.728	0.617	957	1031.1	0.787	12.1
37	0.704	0.632	0.729	0.622	947	1034.4	0.789	12.2
38	0.810	0.622	0.725	0.604	950	1021.4	0.780	13.2
39	0.811	0.629	0.731	0.615	952	1033.9	0.790	13.4
40	0.643	0.576	0.680	0.545	845	946.5	0.723	10.5

In addition, the up-time for testing machines and the draw out machines did not exceed the specified limit. For this reason, it was decided not to build metamodels for those variables.

After observing the throughput, minimum and average columns, it was decided to use the data of the average to build the metamodel. It was also decided to build metamodels for the utilization of the packing personnel and the work-in-process inventory, since their respective limits were exceeded for some experimental runs.

4.4.3.1. Throughput

Figure 4.60 is an output summary from Minitab for the coefficients of the regression and the analysis of variance.

Response Surface Regression: Throughput (Avg.) versus IT, Y, TPT

The analysis was done using coded units.

Estimated Regression Coefficients for Throughput (Avg.)

Term	Coef	SE Coef	T	P
Constant	1033.94	0.8203	1260.430	0.000
IT	56.42	0.5443	103.661	0.000
Y	66.02	0.5443	121.294	0.000
TPT	-0.20	0.5443	-0.369	0.714
IT*IT	2.69	0.5298	5.086	0.000
Y*Y	-0.37	0.5298	-0.703	0.488
TPT*TPT	-0.23	0.5298	-0.436	0.666
IT*Y	4.15	0.7111	5.836	0.000
IT*TPT	0.23	0.7111	0.316	0.754
Y*TPT	0.30	0.7111	0.422	0.676

S = 2.844 R-Sq = 99.9% R-Sq(adj) = 99.8%

Analysis of Variance for Throughput (Avg.)

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Regression	9	206478	206478	22942.0	2835.59	0.000
Linear	3	205973	205973	68657.8	8485.98	0.000
Square	3	227	227	75.6	9.35	0.000
Interaction	3	278	278	92.6	11.45	0.000
Residual Error	30	243	243	8.1		
Lack-of-Fit	5	49	49	9.7	1.26	0.314
Pure Error	25	194	194	7.8		
Total	39	206721				

Figure 4.60 Regression analysis for throughput – 5 testing machines.

The analysis of the coefficients suggested that the model had a quadratic behavior. The metamodel was defined as Equation (4.7).

$$Throughput_{5TM} = 1033.94 + 56.42 * IT + 66.02 * Y + 2.69 * IT^2 + 4.15 * IT * Y \quad (4.7)$$

Where, $Throughput_{5TM}$ is the estimate for the throughput when 5 testing machines are used, IT is the inter-arrival time, and Y is the yield.

It was confirmed by the analysis of variance that the model had a quadratic behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the throughput. Figure 4.61 presents several plots related to the normality and randomness of the residuals, such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals and residuals vs. the order of the data.

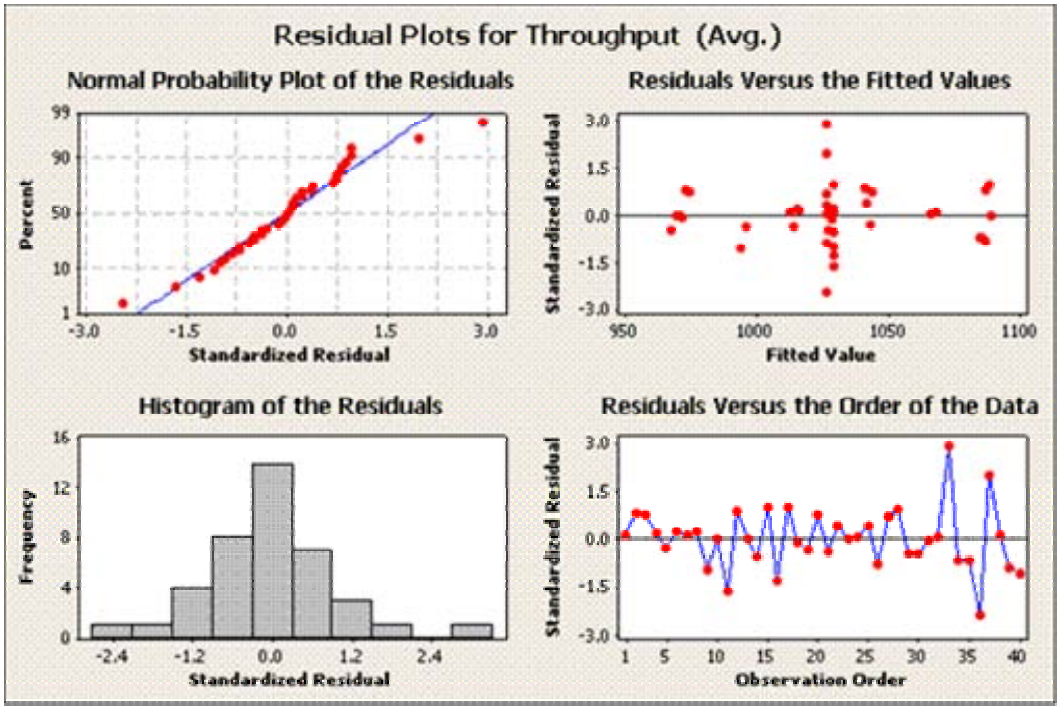

Figure 4.61 Residual plots for throughput – 5 testing machines.

The major concern with regard to the validation was the plot of the residuals vs. the fitted values. It was observed that the variability at the middle point was higher than the rest of the chart. This visual impression was caused by two possible outliers with standard residual values of 2.07 and 3.22. No major reason was identified for the outliers since they seemed to be valid points. It was decided to maintain the two observations as

part of the analysis. Note that at the decision point (constraint defined by the management) with respect to the throughput (1000 units) the variability is minimum, thus it was concluded that the probability for the model rejecting an alternative that was feasible was low.

Normality test and runs test were done in order to gain more confidence with respect to the validation of the assumptions from the analysis of variance. Figure 4.62 and Figure 4.63 shows the results of those tests. It was concluded that both assumptions were validated based on the results of these tests (observe $p > 0.05$) and the explanation provided in the previous paragraph.

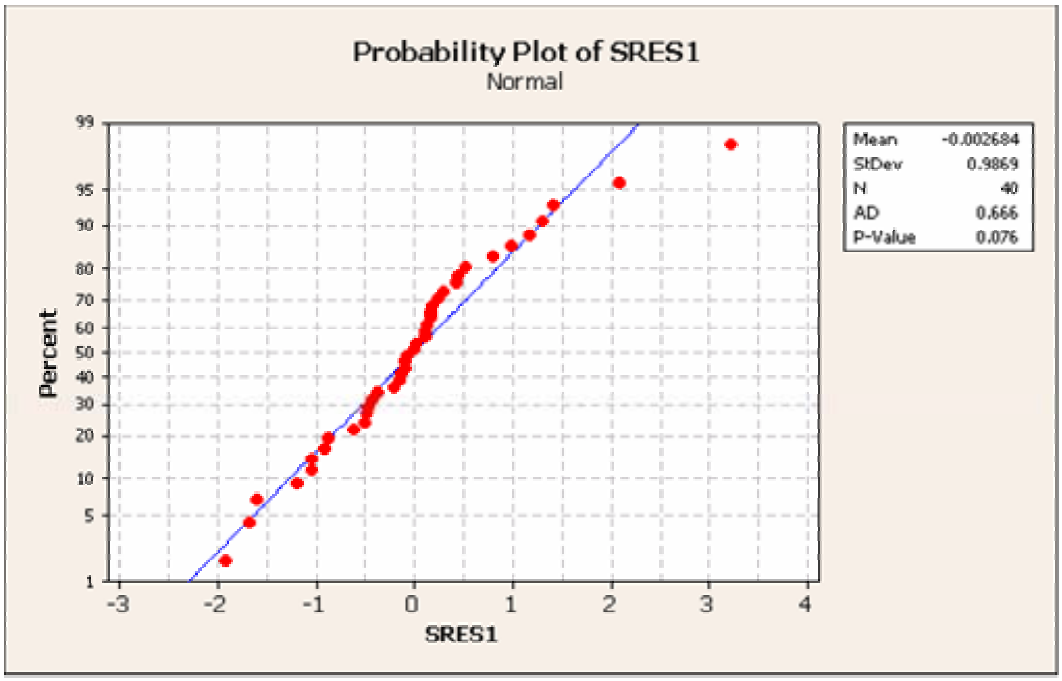

Figure 4.62 Normal probability plot for throughput – 5 testing machines.

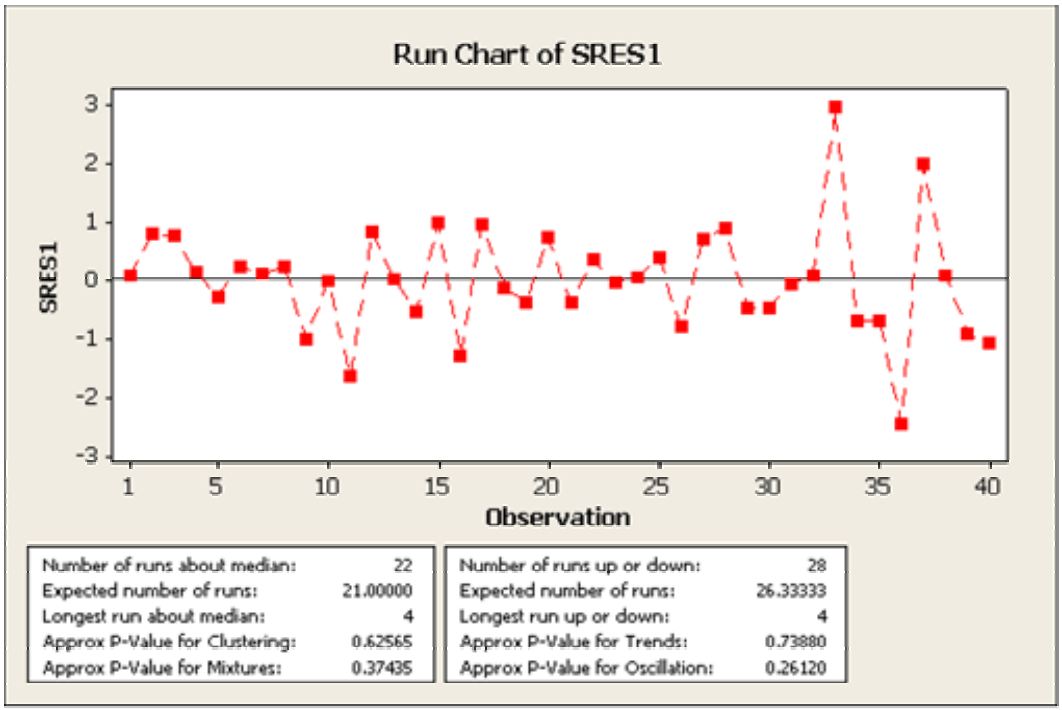

Figure 4.63 Runs test for throughput – 5 testing machines.

Figure 4.64 is a surface plot for the critical factors (inter-arrival time and yield); note the linear relation with respect to the throughput, even though the metamodel has been estimated as a quadratic model. It was observed that the throughput increases as both factors moved toward their respective high levels of the experimental region. Note that the yield was restricted to its high level by definition of the constraint specified by the management of the company. In addition, Figure 4.65 is showing that the inter-arrival time is restricted at its low level of the experimental region based on the throughput constraint of 1000 units. Thus, it was concluded that the testing process time factor was restricted at both sides of the experimental region since its high level was the constraint specified by the management.

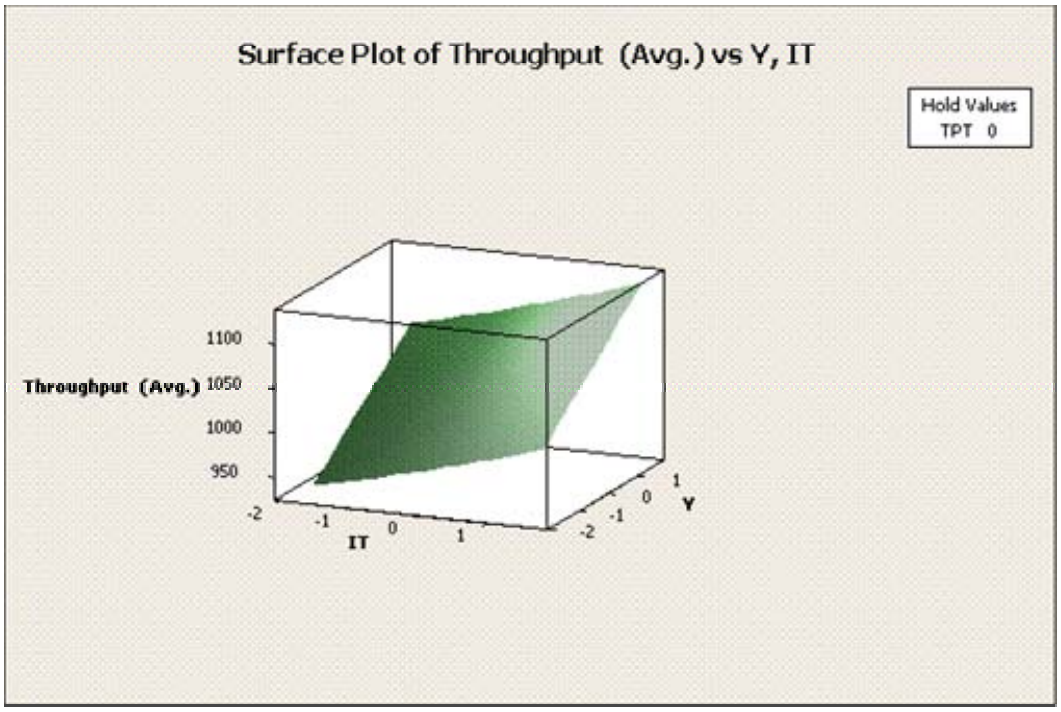

Figure 4.64 Surface plot for throughput vs. inter-arrival time and yield – 5 testing machines.

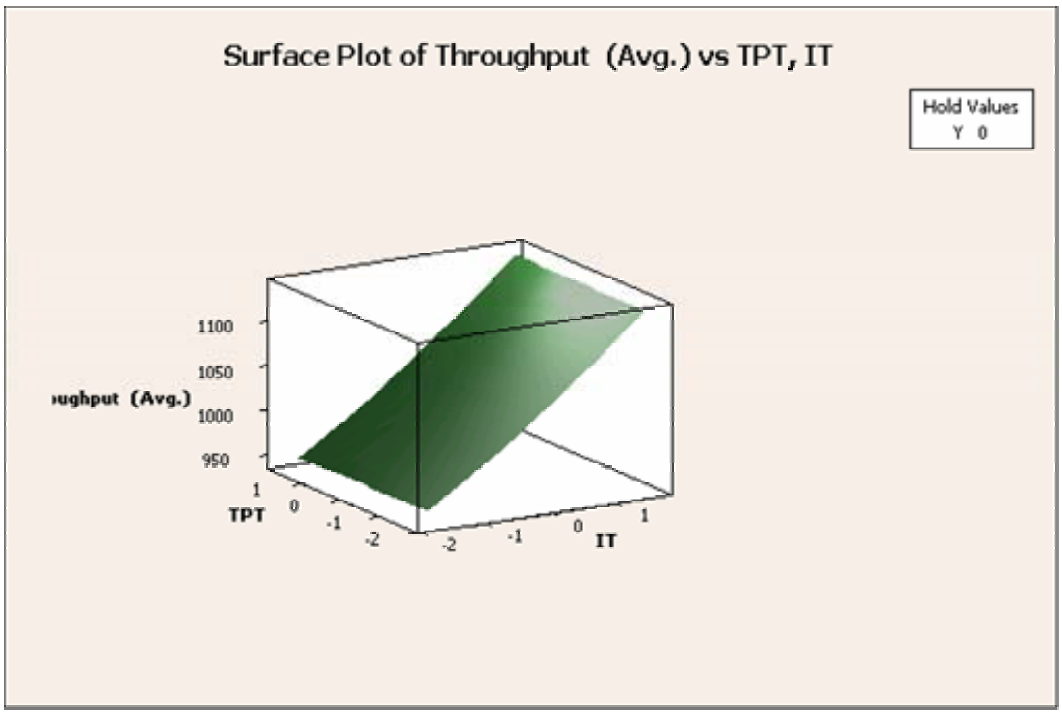

Figure 4.65 Surface plot for throughput vs. inter-arrival time and yield – 5 testing machines.

4.4.3.2.Up-time Packing Personnel

Figure 4.66 is an output summary from Minitab for the coefficients of the regression and the analysis of variance. The analysis of the coefficients suggested that the model had a quadratic behavior. Equation (4.8) is the metamodel for the utilization of the packing personnel.

$$UtilizationPackingPersonnel_{5TM} = 0.789163 + 0.043117 * IT + 0.050407 * Y + 0.0021963 * IT^2 + 0.003187 * IT * Y \quad (4.8)$$

Where, $UtilizationPackingPersonnel_{5TM}$ is the estimate for the labor utilization of the packing personnel when 5 testing machines are used, IT is the inter-arrival time, and Y is the yield. It was confirmed by the analysis of variance that the model had a quadratic behavior. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the up-time for the testing machines.

Response Surface Regression: Utilization Pack versus Block, IT, Y, TPT

The analysis was done using coded units.

Estimated Regression Coefficients for Utilization Packing Personnel

Term	Coef	SE Coef	T	P
Constant	0.789163	0.000610	1294.495	0.000
Block	-0.000575	0.000334	-1.720	0.096
IT	0.043117	0.000404	106.599	0.000
Y	0.050407	0.000404	124.624	0.000
TPT	-0.000233	0.000404	-0.576	0.569
IT*IT	0.002193	0.000394	5.570	0.000
Y*Y	-0.000370	0.000394	-0.940	0.355
TPT*TPT	-0.000194	0.000394	-0.491	0.627
IT*Y	0.003187	0.000528	6.032	0.000
IT*TPT	0.000187	0.000528	0.355	0.725
Y*TPT	0.000312	0.000528	0.591	0.559

S = 0.002114 R-Sq = 99.9% R-Sq(adj) = 99.9%

Analysis of Variance for Utilization Packing Personnel

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Blocks	1	0.000013	0.000013	0.000013	2.96	0.096
Regression	9	0.120497	0.120497	0.013389	2996.19	0.000
Linear	3	0.120180	0.120180	0.040060	8964.92	0.000
Square	3	0.000152	0.000152	0.000051	11.37	0.000
Interaction	3	0.000165	0.000165	0.000055	12.28	0.000
Residual Error	29	0.000130	0.000130	0.000004		
Lack-of-Fit	19	0.000071	0.000071	0.000004	0.63	0.812
Pure Error	10	0.000059	0.000059	0.000006		
Total	39	0.120640				

Figure 4.66 Regression analysis for utilization packing personnel – 5 testing machines.

Figure 4.67 was built with the purpose of validating the assumptions of the analysis of variance related to the normality and randomness of the residuals. This figure presents several plots such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals and residuals vs. the order of the data. The major concern with regards to the validation was the plot of the residuals vs. the fitted values. It was observed that the variability at the middle point and at the left side of the chart was higher than the rest of the chart. This visual impression was caused by two possible outliers with standard residual values of 2.06 and 3.21. No major reason was identified for the outliers since they seemed to be valid points. It was decided to maintain the two observations as part of the analysis. Note that at the decision point (constraint defined by the management) with respect to the utilization of the packing personnel (85% units) the variability is minimum, thus it was concluded that the probability for the model rejecting an alternative that was feasible was low.

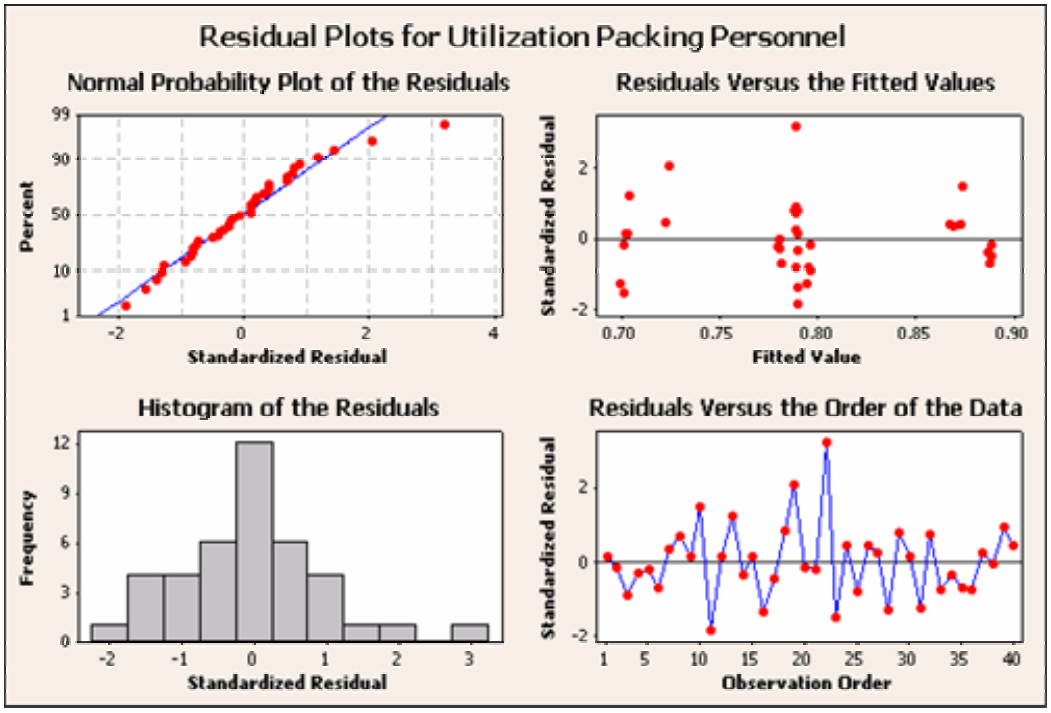

Figure 4.67 Residual plots for utilization of packing personnel – 5 testing machines.

In addition to the explanation offered in the previous paragraph with respect to the assumptions of the analysis of variance, normality test and a runs test were done. Figure 4.68 and Figure 4.69 shows the results of those tests. It was concluded that the

assumption of normality and randomness of the residuals were validated (observe $p > 0.05$).

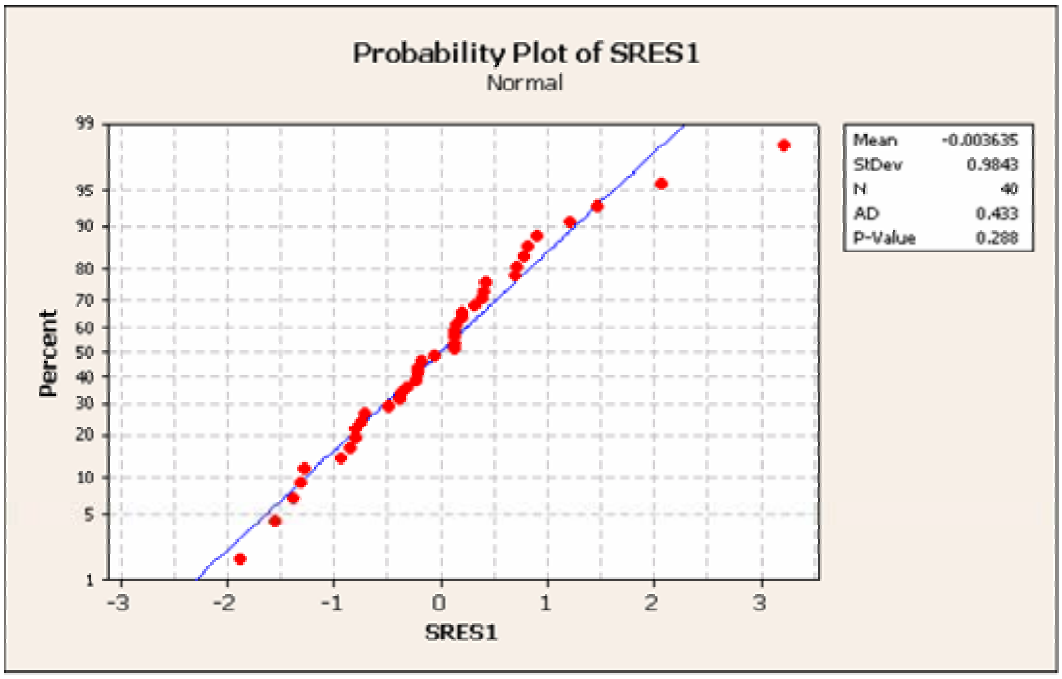

Figure 4.68 Normal probability plot for utilization packing personnel – 5 testing machines.

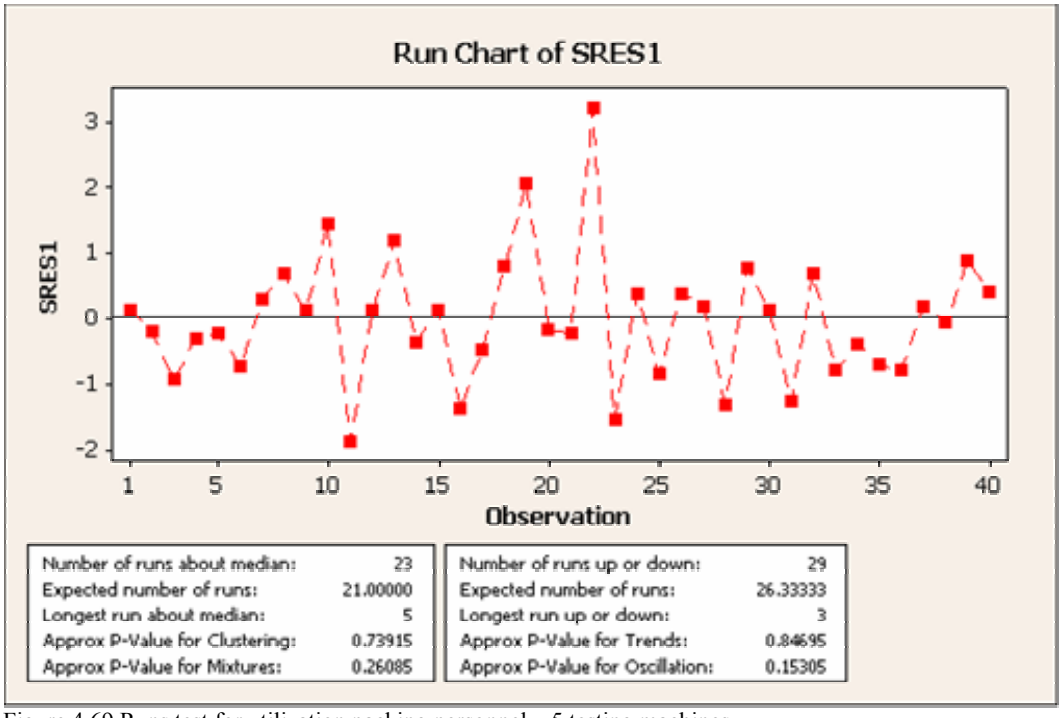

Figure 4.69 Runs test for utilization packing personnel – 5 testing machines.

Figures 4.70, 4.71 and 4.72, are surface plots for the utilization of the packing personnel. Note that the relation with respect to the factors seems to be linear. Nevertheless, the metamodel is a quadratic equation. Note from Figure 4.71, that as the yield approaches its high level on the experimental region, the utilization of the packing personnel tends to be above 85%, which is the constraint specified by the management of the company. The yield is also restricted at the high level of the experimental region as result of the management constraint of 90%. Note from Figure 4.72, that as the inter-arrival time approaches its high level on the experimental region, the utilization tends to be above 85%, thus restricting the yield at that side.

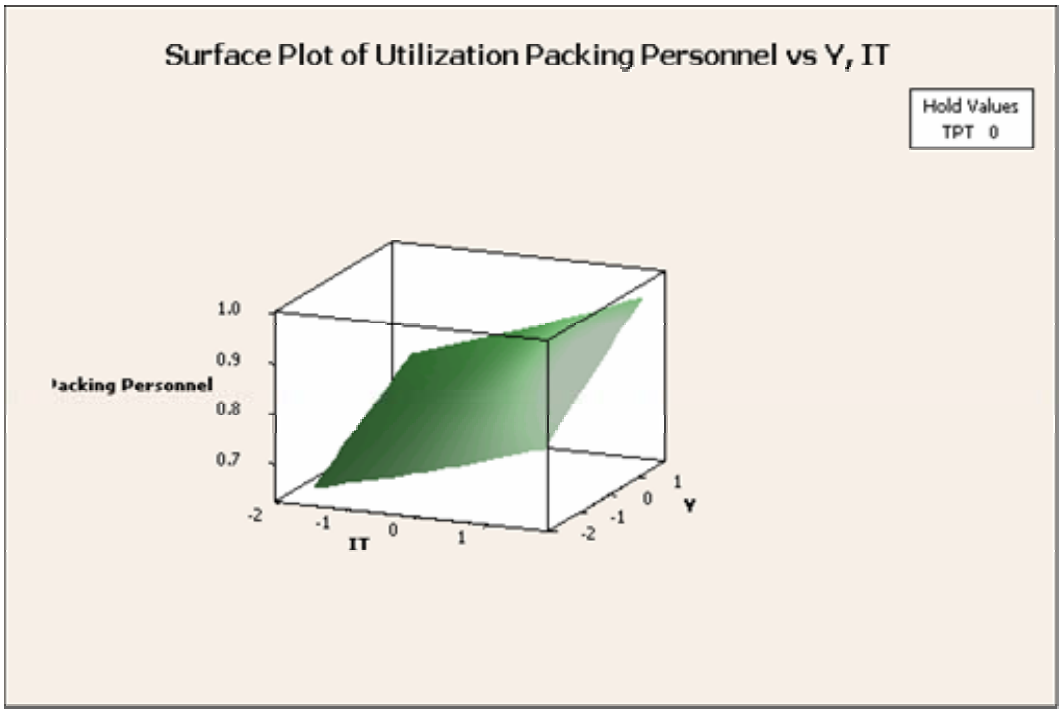

Figure 4.70 Surface plot for utilization packing personnel vs. inter-arrival time and yield – 5 testing machines.

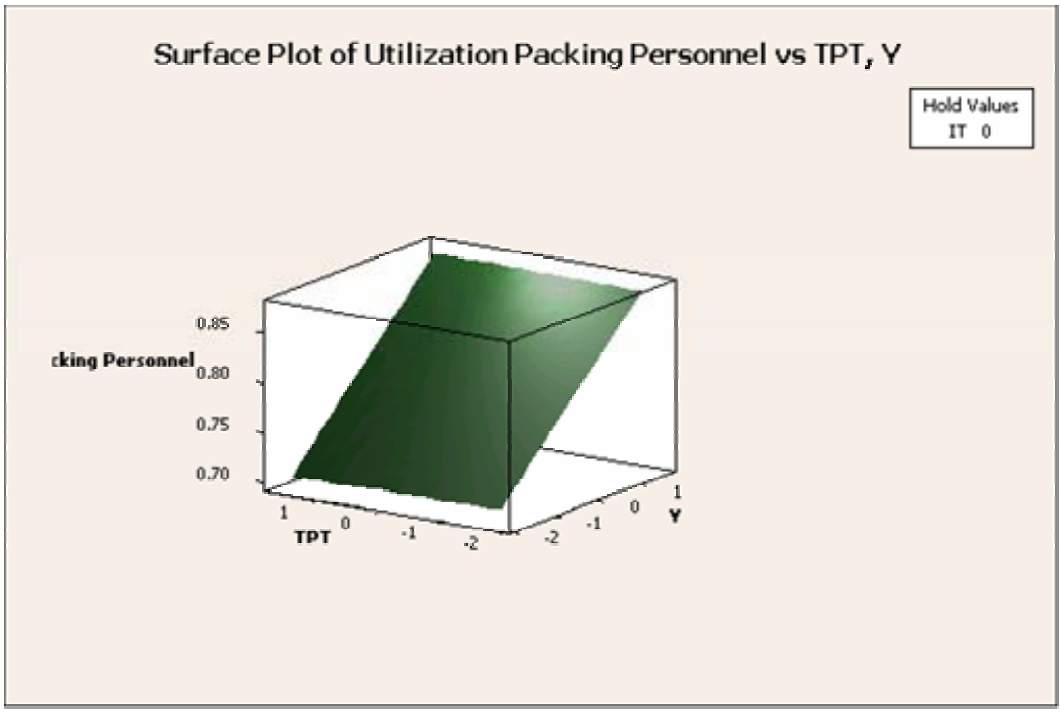

Figure 4.71 Surface plot for utilization packing personnel vs. yield and testing process time – 5 testing machines.

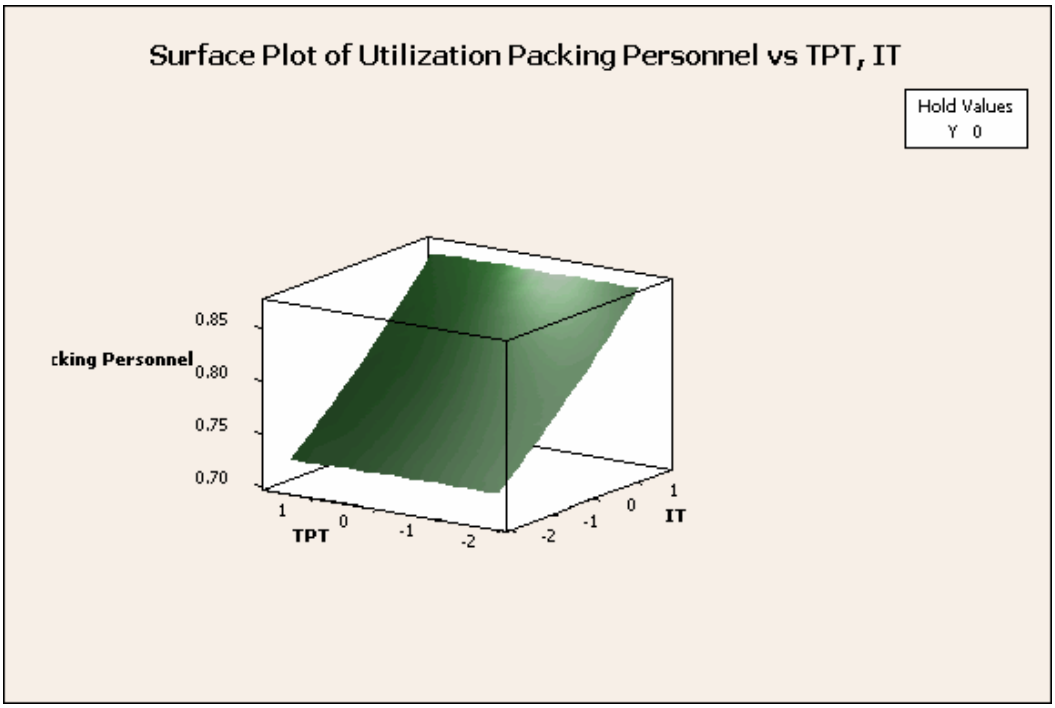

Figure 4.72 Surface plot for utilization packing personnel vs. inter-arrival time and testing process time – 5 testing machines.

4.4.3.1. Work-in-Process Inventory

Figure 4.73 is an output summary from Minitab for the coefficients of the regression and the analysis of variance. It was observed from the analysis of the coefficients that the regression had a quadratic behavior. The metamodel developed is presented in Equation (4.9).

$$\begin{aligned} WIP_{5TM} = {} & 12.1428 + 1.422 * IT + 1.182 * Y - 0.5623 * TPT + 0.2343 * IT^2 \\ & + 0.1813 * Y^2 + 0.0929 * TPT^2 + 0.375 * IT * Y \\ & - 0.0875 * IT * TPT + 0.05 * Y * TPT \end{aligned} \tag{4.9}$$

Where, WIP_{5TM} is the estimate for the work-in-process inventory when 5 testing machines are used, IT is the inter-arrival time, Y is the yield, and TPT is the testing process time.

Response Surface Regression: WIP versus Block, IT, Y, TPT

```
The analysis was done using coded units.

Estimated Regression Coefficients for WIP

Term         Coef  SE Coef        T      P
Constant  12.1428  0.02192  553.997  0.000
Block     -0.0100  0.01202   -0.832  0.412
IT         1.4220  0.01454   97.784  0.000
Y          1.1820  0.01454   81.283  0.000
TPT       -0.5623  0.01454  -38.667  0.000
IT*IT      0.2343  0.01416   16.552  0.000
Y*Y        0.1813  0.01416   12.805  0.000
TPT*TPT    0.0929  0.01416    6.562  0.000
IT*Y       0.3750  0.01900   19.736  0.000
IT*TPT    -0.0875  0.01900   -4.605  0.000
Y*TPT      0.0500  0.01900    2.631  0.013

S = 0.07600   R-Sq = 99.8%   R-Sq(adj) = 99.8%


Analysis of Variance for WIP

Source          DF   Seq SS   Adj SS   Adj MS        F      P
Blocks           1    0.004    0.004   0.0040     0.69  0.412
Regression       9  106.844  106.844  11.8716  2055.21  0.000
  Linear         3  102.033  102.033  34.0109  5887.95  0.000
  Square         3    2.399    2.399   0.7997   138.45  0.000
  Interaction    3    2.412    2.412   0.8042   139.22  0.000
Residual Error  29    0.168    0.168   0.0058
  Lack-of-Fit   19    0.099    0.099   0.0052     0.76  0.706
  Pure Error    10    0.068    0.068   0.0068
Total           39  107.016
```

Figure 4.73 Regression analysis for WIP – 5 testing machines.

It was confirmed by the analysis of variance the quadratic behavior of the model. The lack of fit test as well as the R^2_{adj} confirmed that the regression equation or metamodel was a good estimator of the work-in-process inventory. Figure 4.74 presents several plots related to the normality and randomness of the residuals, such as: normality plot of the residuals, residuals vs. fitted values, histogram of the residuals and residuals vs. the order of the data. The plots suggested the validation of the assumptions for the analysis of variance. A normality test and runs test were performed in order to have robust evidence on which to base the conclusion with respect to the assumptions. Figure 4.75 and Figure 4.76 shows the results of those tests. The normality and randomness assumptions were validated (observe *p-values* > *0.05*).

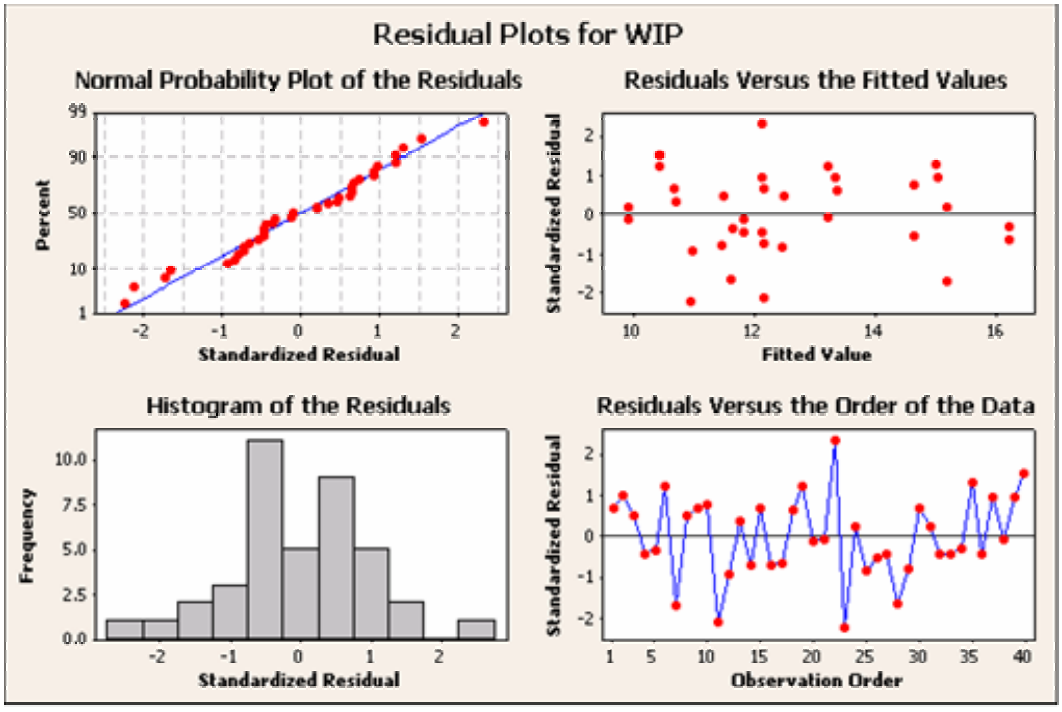

Figure 4.74 Residual plots for WIP – 5 testing machines.

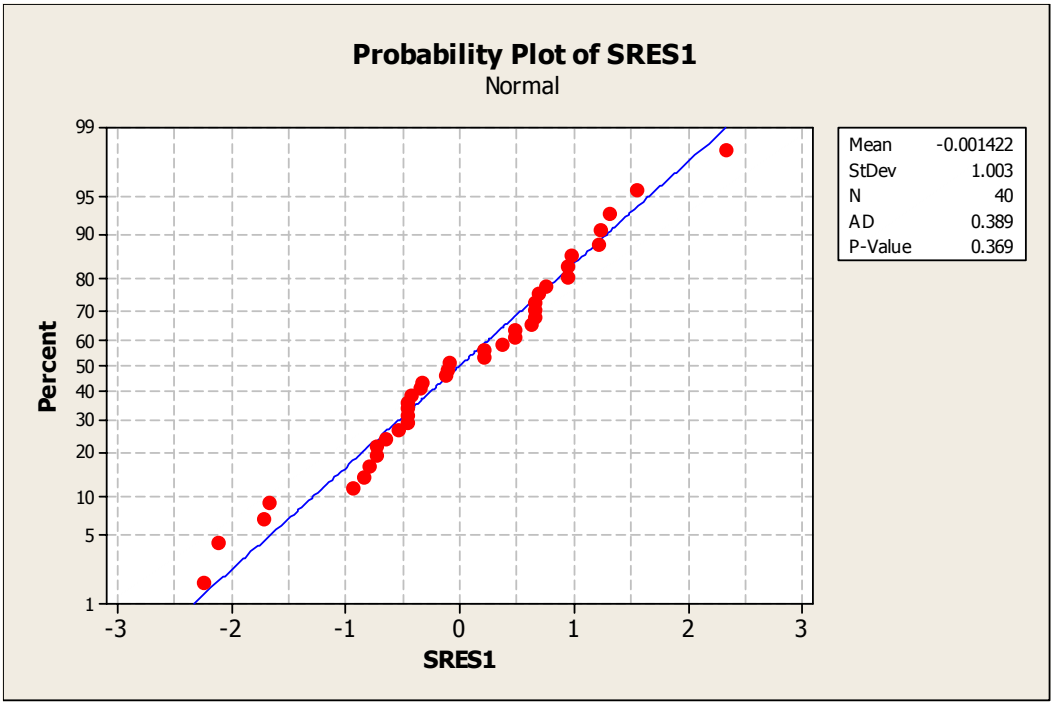

Figure 4.75 Normal probability plot for WIP – 5 testing machines.

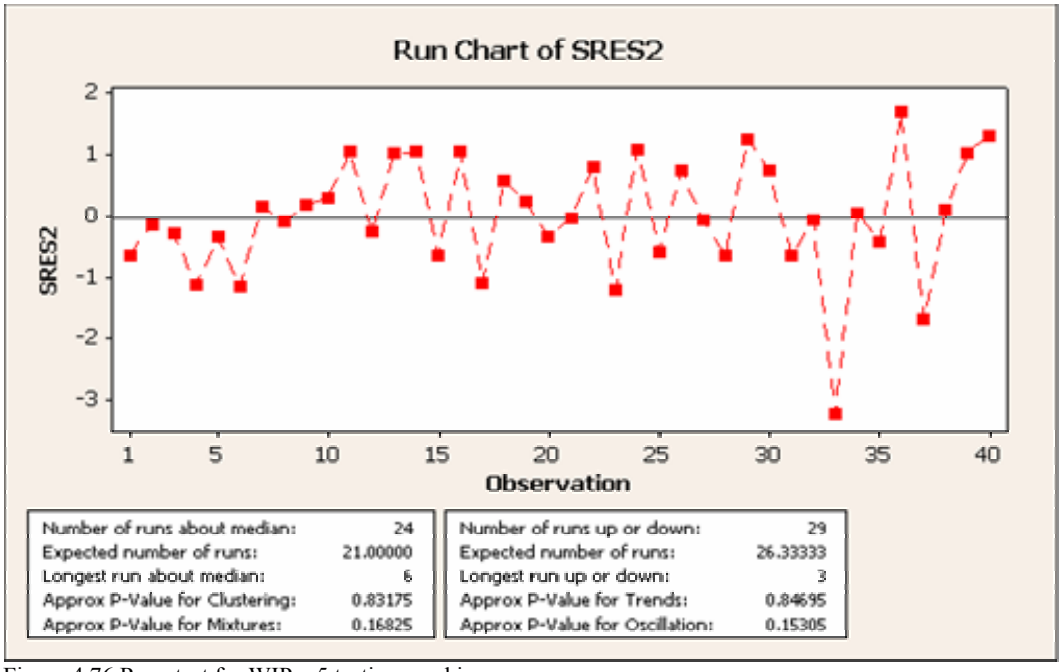

Figure 4.76 Runs test for WIP – 5 testing machines.

Figures 4.77, 4.78, and 4.79 are surface plots for the work-in-process inventory. Note the quadratic relationship with respect to the factors of the metamodel. It is important to mention that this metamodel was built for the purpose of doing the

optimization (objective function) since all values were within the acceptable region with respect to the constraint specified by the management of the company. Note that the surface response was always less than 22 units.

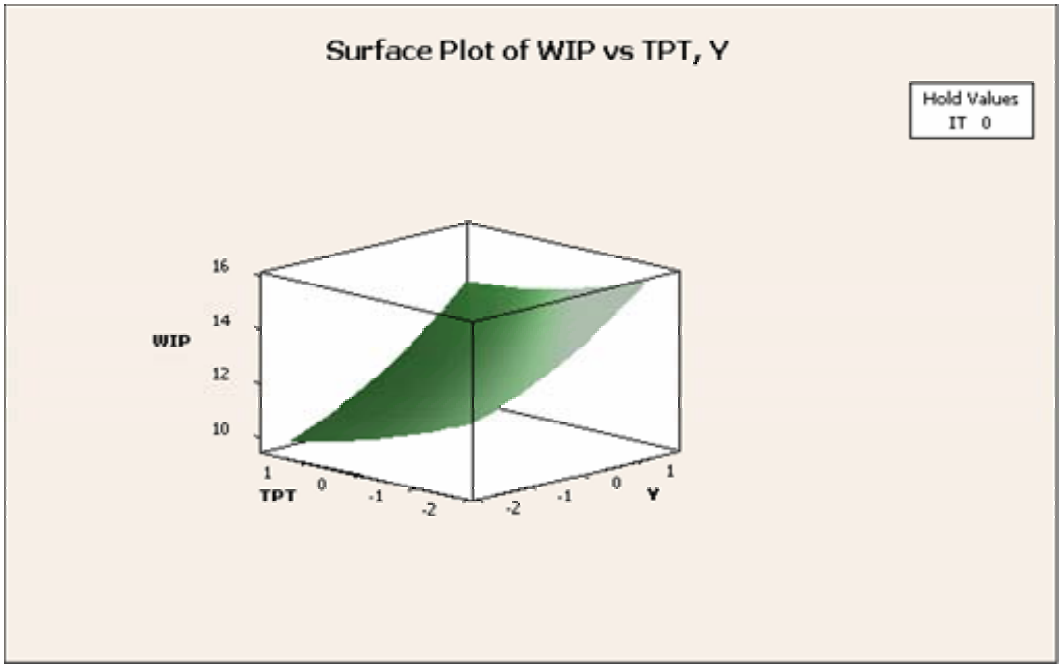

Figure 4.77 Surface plot for WIP vs. yield and testing process time – 5 testing machines.

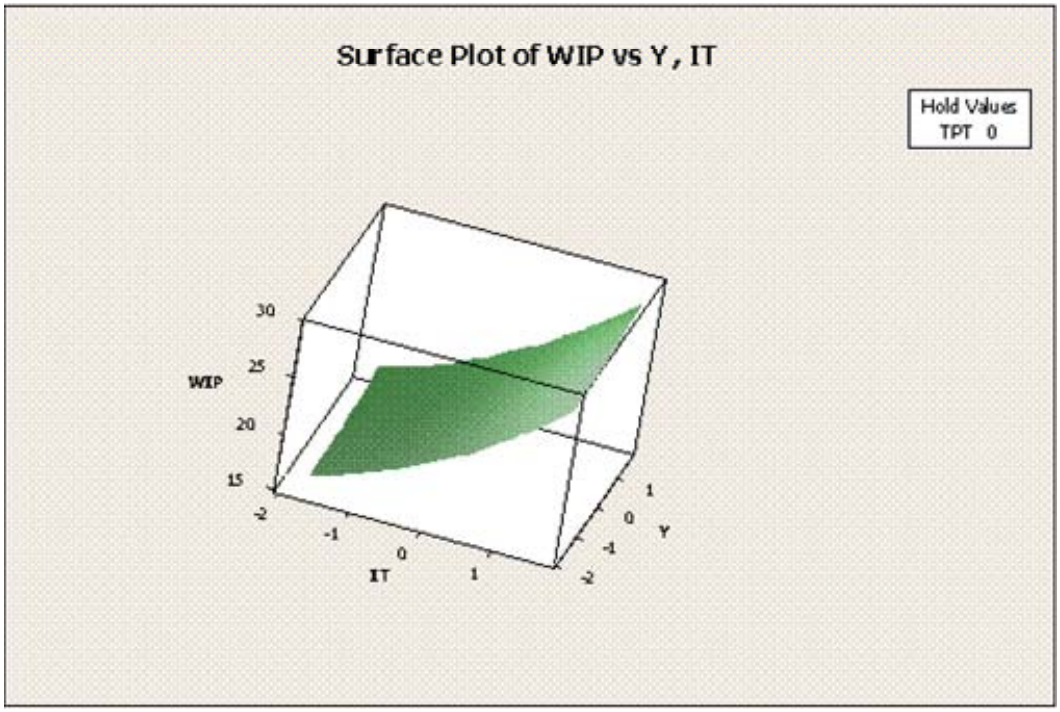

Figure 4.78 Surface plot for WIP vs. inter-arrival time and yield – 5 testing machines.

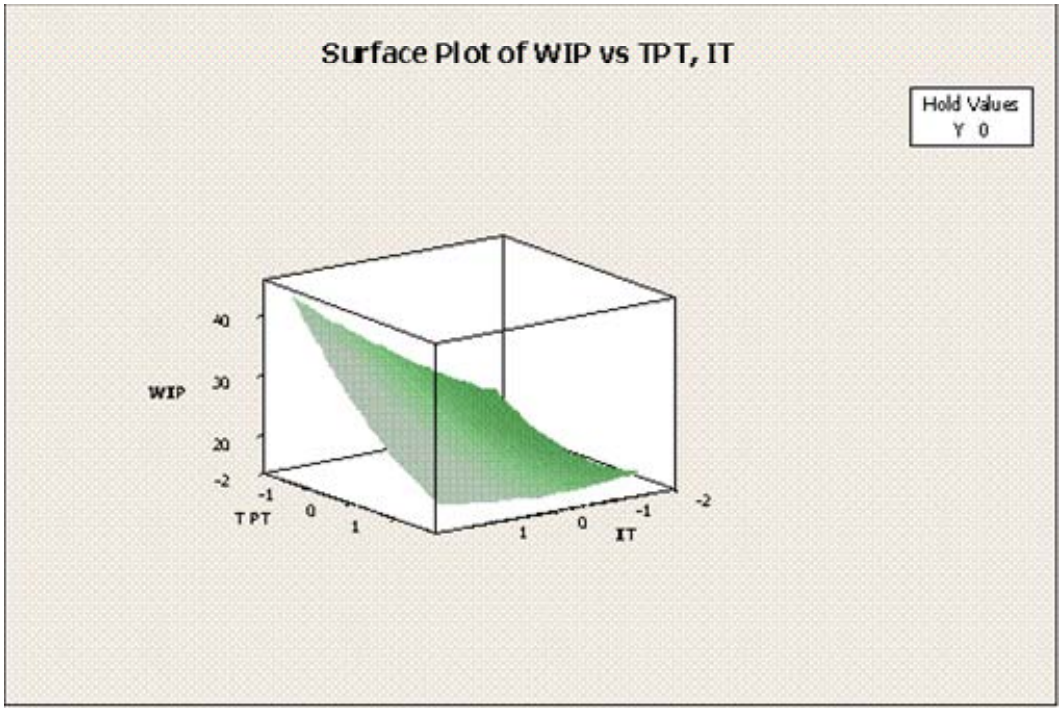

Figure 4.79 Surface plot for WIP vs. inter-arrival time and testing process time – 5 testing machines.

4.5. Optimization Tool

This section explains the results from the validation for the optimization tool. The main objective of the tool was to determine the number of machines that were required to meet the expected production as well as the constraints specified by the management of the company, given a combination of the input parameters yield and testing process time.

The tool was developed using Microsoft's Excel Solver and Visual Basic for Applications (VBA) mainly due to the practicality and inexpensive use of these tools from the end user point of view. In addition, it was already learned from the literature review that Microsoft Excel Solver employs the Generalized Reduced Gradient algorithm which could solve smooth non-linear optimization problems up to 200 decision variables and 100 constraints. So, taking advantage of this algorithm became obvious since it was readily available to the end user, and the mathematical programming models (3, 4, and 5 testing machines) that this project was pretending to solve were of the smooth non-linear type with only 3 decision variables and 3 constraints.

Basically, the end user enters the values for the yield and testing process time, the optimization tool selects then by means of Microsoft Visual Basic for Applications the optimization model (for 3, 4 or 5 testing machines) and Microsoft Excel Solver uses the Generalized Reduced Gradient (GRG) method as implemented in an enhanced version of Lasdon and Waren's GRG2[13] code to find the optimum within the search region. The search variable is the inter-arrival time.

The validation of the optimization tool was done mostly by focusing on three subjects: (1) predicting the accuracy of the metamodels vs. the simulation output, (2) the capability of the experimental region to contain a feasible solution or local optimum with respect to the combination of possible values for the yield and testing process time input parameters, and (3) the capability of the experimental region to contain the global optimum. In particular, the first two subjects were key to show the validity of the Generalized Reduced Gradient method as well as the usefulness of Microsoft Excel Solver for the optimization stage of this simulation project

4.5.1. Accuracy of the Metamodels

This section explains the results of the validation with respect to the metamodels accuracy as a tool to predict the point estimate. Table 4.28 presents the results for each validation run or scenario, and Table 4.29 presents the accuracy (error %) of the metamodels vs. the simulation outputs or responses. Note from Table 4.29 that the accuracy of the metamodel is within the allowable error for each of the performance measures of interest under each scenario. It was then concluded that the metamodels were validated.

[13] This algorithm was developed by Leon Lasdon, of the University of Texas at Austin, and Allan Waren, of Cleveland State University. As per Frontline Systems, Inc., the company who developed the Solvers/Optimizers in Microsoft Excel, Lotus 1-2-3 and Quattro Pro, the GRG2 code has been proven in use over many years as one of the most robust and reliable approaches to solving difficult NLP problems.

Table 4.28 Optimization tool and simulation results for validation purposes.

		Input		Output	Simulation Model				Optimization Tool			
Scen. #	Opt. Model	Yield (%)	Testing Proc. Time (%)	Inter-arrival Time	Through.	Up-time Testing Mach.	WIP	Util. Pack. Pers.	Through.	Up-time Testing Mach.	WIP	Util. Pack. Pers.
1	3 TM	90.0	70.0	0.465	1096.2	94.5	19.7	N/A	1090.4	94.3	18.6	N/A
2	4 TM	81.3	85.0	0.450	1021.2	86.7	13.7	N/A	1019.7	86.4	13.7	N/A
3	5 TM	72.6	100.0	0.405	1016	N/A	15.5	77.6	1013.7	N/A	14.3	77.4

Table 4.29 Accuracy (error %) of the metamodel vs. the simulation.

		% Error			
Scen. #	Opt. Model	Through.	Up-time Testing Mach.	WIP	Util. Pack. Pers.
1	3 TM	0.53%	0.21%	5.58%	N/A
2	4 TM	0.15%	0.35%	0.00%	N/A
3	5 TM	0.23%	N/A	7.74%	0.26%
		2.50%	1.00%	10.00%	1.00%
		Allowable Error			

4.5.2. Validation of the Experimental Region

This section explains the results of the validation with respect to the capability of the experimental region to contain a feasible solution (local optimum). Table 4.30 shows that scenario #1 has no feasible solution for 3 testing machines; instead, a feasible solution was obtained for 4 testing machines. Table 4.31 shows that scenario #2 has no feasible solution for 4 testing machines; instead, a feasible solution was obtained for 5 testing machines. These were the expected results.

Table 4.30 Scenario #1 – Experimental region.

		Optimization Tool	
	Constraints	3 Testing Machines	4 Testing Machines
Throughput	>= 1000	1000.0	1063.7
Up-time Test. Mach.	<= 95%	**96.7**	77.6
WIP	<= 22	**24.6**	12.6
Scenario #1			

Table 4.31 Scenario #2 – Experimental region.

		Optimization Tool	
	Constraints	4 Testing Machines	5 Testing Machines
Throughput	>= 1000	**971.4**	1113.1
Up-time Test. Mach.	<= 95%	**96.6**	N/A
WIP	<= 22	**27.8**	16.1
Util. Pack. Pers.	<= 85%	N/A	85.0
Scenario #2			

The conclusion from the above two tables was that the experimental region (in terms of the yield and testing process time factors) for a particular number of testing machines, had the capability to contain the optimum for a particular combination of yield and testing process time if in reality a feasible solution was within that particular number of testing machines.

Validating the experimental region was important because it guaranteed the right selection of the number of testing machines required for a particular combination of yield and testing time factors. It also showed the validity of the Generalized Reduced Gradient method as well as the usefulness of Microsoft Excel Solver for the optimization stage of this simulation project.

4.5.3. Validation of the Global Optimum

It was observed from tables 4.28 and 4.30 that the optimum was found before any of the constraints reached their respective limits. The main reason for this behavior was that the distance between the levels (low and high) of the inter-arrival time factor was not large enough for allowing the global optimum to be found; thus the optimization tool found only a feasible solution which was in fact the local optimum within the experimental region.

The search was not done outside the limits of the inter-arrival time factor because there was no evidence that the metamodel would still explain the behavior of the simulation response. One way to address this issue in the future is by expanding the limits for the inter-arrival time factor, and then performing a new experimental design with the objective of developing new metamodels. Nevertheless, the results were still valid in terms of the number of machines required for a particular combination of yield and testing process time factors.

4.6. Comparison of Methodologies

This section pretends to compare the results obtained by the scientific approach followed for this project (2) vs. the trial and error approach followed for the previous project (1). The comparison was done for the particular combination in which the yield was set to 90% and the testing process time was set to 70% of the value of the actual system, which were the suggested conditions presented to the management of the company during the previous project. For both scenarios, the goal for the performance measure throughput was a total of 3000 units per month and all constraints were met. Table 4.32 presents the results in terms of number of personnel and equipment required. Inventory related data was not included because the difference was negligible between both methods.

Table 4.32 Comparison actual vs. previous project.

		Previous Project (1)	Actual Project (2)	Benefits ($)
Testing Area	Dielectric Machines	1	1	$0
	Testing Machines	4	3	$350,000
	Testing Personnel	4	2	$30,576
Draw Out Area	Draw Out Machines	2	3	($50,000)
	Assembly Personnel	2	4	($26,416)
	Final Inspectors	2	2	$0
Packing Area	Packing Personnel	2	2	$0

It was concluded from the above table that the savings under method (2) as compared to method (1) were $300,000 for capital expenditures and $4,160 per year for labor cost. There were two main differences in terms of both scenarios: (1) labor balancing between the testing and draw out areas and (2) the maximum throughput that could be obtained from the systems. With respect to the labor balancing, scenario #1 balanced the labor by locating Inspection I at the testing area, while in scenario # 2 the labor remained unbalanced by maintaining Inspection I at the draw out area. The second major difference was that the maximum throughput for the system under scenario #1 was 3130 units, while scenario #2 could reach a maximum of 3270 units. This in fact explained the addition for the third draw out machine. This extra capacity would have given the company another benefit of 140 units or the equivalent of $490,000 in extra sales per month. It would have been management decision to decide if the investment on the extra capacity was worth it. Under both scenarios, the capacity limitation turned out to be the number of testing machines.

Information that could help the management on decision making is the probability for achieving the throughput value (or more) at which the system was setup. Figure 4.80 shows the probability curve for the monthly throughput under scenario #2 for 1 shift. The goal was to set a system that could achieve 3000 units monthly (1000 units per shift,

having 3 shifts). It was observable from the chart that the probability for reaching 3270 units or 1090 per shift ($P(X \geq 1090)$) was ±60%. It was understood that the probability for reaching 3130 units or 1043 per shift ($P(X \geq 1043)$) was ±50% for scenario #1, while for scenario #2 the probability was ±95%. In fact, the probability for not achieving 1000 units monthly per shift was almost 0% for scenario #2, which met the expectation set earlier about achieving a total monthly throughput of 3000 units most of the time. This could be a valuable piece of information for deciding whether to invest in the extra capacity or not. Note that on the eventuality that the management decision is not to invest in the extra capacity, the capital cost savings between methods (1) and (2) are $350,000 instead of $300,000. In addition, the possibility for reducing labor from scenario #2 is worth trying with the simulation model. Of course, this will require revisiting the metamodels.

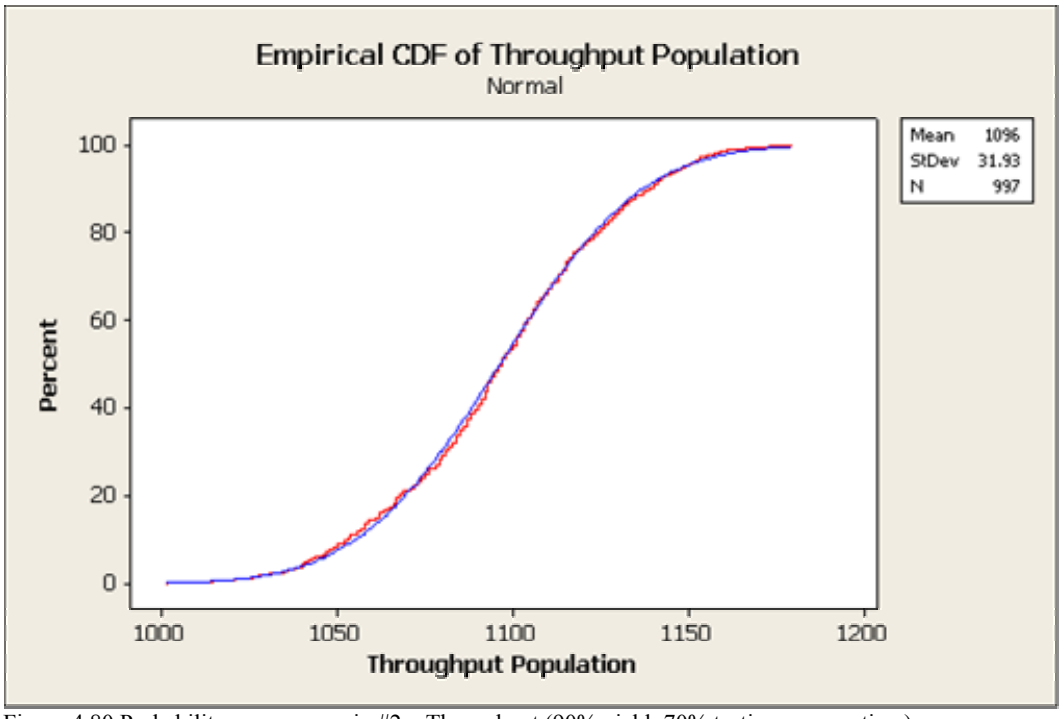

Figure 4.80 Probability curve scenario #2 – Throughput (90% yield; 70% testing process time).

Table 4.33 shows the average and maximum number of units at each buffer. It was concluded that for space requirements the management should plan around the

maximum number of units. If less space was required, then it was imperative to look into reasons for the variability at the testing area.

Table 4.33 Buffer requirements.

Buffer	Average (units)	Maximum (units)
Dielectric	0.08	0.12
Testing	8.80	51.67
Adjustment	0.07	0.09
Inspection I	0.05	0.11
Draw Out	0.11	0.18
Final Inspection	0.32	0.42
Packing	0.89	2.08
WIP	19.70	62.98

Finally, it was concluded that the scientific method followed in this project provided better results than the trial error approach followed in the previous project.

4.7. Summary

This chapter presented the analysis and results of the scientific approach described in Chapter III. It covered the sensitivity analysis, experimental designs, metamodels, and the validation of the optimization tool.

The warm-up period was validated and the critical input parameters (experimental factors) with respect to the performance measures of interest were identified by means of the sensitivity analysis. Several experimental designs were run and results analyzed with the main objective of reducing the number of critical factors. The model was simplified by defining the values at which to fix input parameters that were taken out from the critical list as the problem was being reduced. Regression analysis was done to develop the metamodels which were used to define the mathematical models for optimization purposes. Three optimization models (3, 4, and 5 testing machines) were developed and validated. The results showed the validity of the General Reduced Gradient method as well as the usefulness of Microsoft Excel Solver

The following chapter (V) presents the conclusions and recommendations with respect to this chapter. Conclusions will be presented in terms of the practical application that this project was trying to solve and in terms of the academic interest for simulation output analysis and optimization. Recommendations will be presented in terms of opportunities for future research.

CHAPTER V. Conclusions and Recommendations

This chapter explains the conclusions and recommendations related to the execution of this project. Conclusions are provided with respect to the objectives defined in section 1.5, while recommendations are related to opportunities for future research.

5.1. Conclusions

Conclusions are presented in terms of the practical application of computer simulation for the system under study, as well as the academic interest on simulation output analysis for systems with multiple response variables and restrictions.

5.1.1. Practical Application

The system under study was the assembly line for a new product manufactured by a multinational company, in particular the testing, final inspection and packing areas. Due to the market acceptance for the new product, the management of the company was interested in defining the arrangement needed in order to increase the monthly throughput from 800 units to 3000 units with the minimum investment in terms of capital and operational costs. Specifically, they were interested in knowing the amount of equipment needed and personnel requirements. Restrictions were specified by management for the maximum allowable personnel utilization and equipment up-time, as well as for the work-in-process inventory.

In order to provide an answer to the management, it was important to focus first on the line balancing between the testing and the draw out/final inspection areas. An experimental design was run an analyzed to address this issue, mainly focused on the location of Inspection I. It was decided to leave Inspection I at the draw out area, even though this represented running the line on an unbalanced fashion.

The level at which to set the number of dielectric and draw out machines, as well as the number of testing, assembly, final inspection and packing personnel, was defined after a series of sequential experimental designs. It was concluded that 1 dielectric machine and 3 draw out machines were required to achieve 3000 units consistently. With

respect to the labor requirements, it was defined that 2 testing personnel, 4 assembly personnel, 2 final inspectors and 2 packers were required.

It was concluded that the number of testing machines was dependable upon the combination of yield and testing process time, ranging from 3 to 5 machines. An optimization tool was developed so that management could define the number of machines. The tool was developed using Microsoft Excel Solver and Visual Basic for Applications which was of practical value for the end user.

For the specific combination of 90% yield and 70% testing process time, it was determined that 3 testing machines were required. Space requirements in terms of buffers were determined. The probability curve for the throughput was defined; it was found that the probability for achieving a monthly throughput of 3000 units (1000 units per shift having 3 shifts) was almost 100%.

5.1.2. Academic Interest

This project was a continuation of a previous project in which the yield and testing process time were fixed to 90% and 70%, respectively, followed by a trial and error approach in order to provide an alternative. As a result, the capacity for understanding the model under the whole spectrum of alternatives defined by the possible combinations of yield and testing process time was lost. In addition, there was no guarantee that the system suggested was the optimum or best alternative in terms of capital and operational costs.

The main academic interest of this project was to understand the behavior of the system by means of sensitivity analysis, experimental designs, regression analysis for the construction of metamodels, and optimization. These tools as well as the appropriate statistical analysis for the simulation responses were key ingredients for the success of this project. The optimization tool was developed using Microsoft Excel Solver and Visual Basic for Application, mainly due to their availability and practicability for the end user. The optimization problem that this project pretended to solve was of the smooth non-linear type which the standard Microsoft Excel Solver had the capability to solve by

means of the Generalized Reduced Gradient method. Another major academic interest was the comparison of both methods (trial and error vs. scientific approach) for the particular combination of 90% yield and 70% testing process time.

The use of sensitivity analysis showed to be of great value at the time of defining critical input parameters that were included in the experimental designs. In addition, it was also an important tool for the determination of the experimental levels for those critical factors. Finally, it also helped on reducing the problem.

The use of experimental designs was important for reducing the problem and understanding the relationship between the critical factors and the performance measures of interest. It was also the base for the regression analyses and the construction of the metamodels. Performing 2^{k-p} fractional factorial design was an important tool for factor-screening purposes at the beginning of the experimental phase. Nevertheless, other type of experimental design may have been more appropriate. In this case in particular, the statistical analysis of the experiment was sacrificed in favor of the number of simulation replicates and computer run time. The analysis was done by means of graphical plots since the factors being screened were qualitative and discrete. 2^k factorial designs showed also to be useful for setting the draw out testing area in terms of the number of machines and personnel required.

Once the model was reduced, the focus of the project was on understanding the behavior at the testing area. It was understood that typical simulation optimization tools were not of practical value at this stage, since management had no certainty for the values at which to set the yield and the testing process time. This forced the analysis to focus on understanding the behavior of the model for all possible combinations. It was decided to perform three experimental designs, one for each number of testing machines (3, 4 and 5). The major concerns were (1) to define the experimental region so that feasible solutions were guaranteed for any possible combination of yield and testing process time, and (2) the presence of curvature. Central composite designs were run in which the experimental region for 3 machines was contained inside the experimental region for 4 machines, and 4 machines was contained inside the experimental region for 5 machines.

This approach guaranteed a feasible solution for all possible combinations. The curvature concern was also addressed since the base for the central composite design is a quadratic model.

Metamodels were built from the results of the central composite experimental designs. Mathematical programming models were defined and the Generalized Reduced Gradient method was employed for optimization purposes. The accuracy of each model was verified by means of the computer simulation output. It was demonstrated that the optimization tool had the capability to find a feasible solution or local optimum for any combination of yield and testing process time within the experimental region. It was shown that the tool felt short for finding the global optimum. In order to do so, it was concluded that the experimental region must have been wider with respect to the inter-arrival time factor. This could be a matter for future research. Nevertheless, this was of no major concern for the practical problem that this project pretended to address since feasible solutions were being guaranteed.

It was shown that the results obtained from the scientific approach followed during this project rendered better results than the trial and error approach followed during the previous project.

The approach followed (mathematical programming) for understanding the behavior of the system for all combinations of yield and testing process time, and developing the optimization tool using Microsoft Excel Solver and Visual Basic for Applications proved to be of great value for the practical problem being addressed with this project. It is the understanding of the author, that this approach has applications for other practical situations in which (1) the end user does not have the resources for the application of any of the simulation optimization methods frequently found in the simulation literature, and (2) there is uncertainty with respect to the levels at which to fix some of the critical input parameters. Once the metamodels are developed and the optimization model (mathematical programming) built with Microsoft Excel Solver, the end user can practically study all possibilities for decision making purposes without investing on extensive simulation studies.

It is important to mention that there is a trade-off between understanding the behavior of all possible combinations for the critical factors and the cost of developing such a tool. In the case of this project, it was relatively easy since the problem was already reduced to three quantitative critical factors (yield, testing process time, and inter-arrival time) for each number of testing machines. Note that as the number of factors increases, defining the experimental region become more difficult. In addition, the experimental region may tend not to fit a quadratic model, which complicates the analysis even further.

The value of Microsoft Excel Solver for simulation optimization purposes was demonstrated in this project. Its usefulness was based on the fact that Microsoft's tools are commonly used in today's business world, and modeling an optimization problem with Microsoft Excel Solver is relatively easy to do. In addition, the Excel Solver employs algorithms that have been widely used in the optimization arena. The standard Microsoft Excel Solver uses known algorithms, as the Generalized Reduced Gradient method, for solving optimization problems of some degree of complexity, most probably of the complexity range for many simulation optimization problems.

Finally, the impact that the optimization algorithms used by Microsoft Excel Solver may have on solving a mathematical programming model, becomes of interest as the number of critical factors increases as well as the complexity of the model itself.

5.2. Recommendations

This section presents recommendations for future research. The order in which they are presented is of no relevant importance.

- The optimization tool should be further improved by expanding the experimental region for the inter-arrival time in order to find the global optimum for any combination of yield and testing process time.
- The optimization tool should provide information related to all performance measures, including the maximum number of units in each buffer. In addition, the probability curve for the throughput should also be mathematically

modeled for at least increments of ten percent. These whole set of information completely allows the end user not having to depend on the simulation model for a particular combination of yield and testing process time of interest.

- A comparative research between the optimization tool developed for this application and any of the methods typically used for simulation optimization should be done. This research could contribute to the validation of the methodology followed in terms of using mathematical programming and employing the Generalized Reduced Gradient method as the optimization algorithm.

- The complexity of this project was reduced by setting the throughput goal around 3000 units per month. In the eventuality that management has serious concerns with those projections, designing only for the case of 3000 units may have no practical value. This raises the concern for understanding the behavior of the system for the possible values of throughput. This brings back the whole complexity of the project since practically no input parameter could be set to a fix value. Pretending to solve such a system may be difficult but worth trying from the academic point of view.

- The complexity of the research discussed in the previous point may be reduced by fixing the yield and the testing process time, but allowing the throughput to be variable within a range. This research is of less complexity than the previous one but still of practical value as well as an opportunity for integrating mathematical programming with the corresponding optimization algorithms employed by Microsoft Excel Solver.

Finally, the above research opportunities are motivated by the practical application that they have in the industry and the academic interest in simulation optimization.

BIBLIOGRAPHY

Andradóttir, S. 1998. A Review of Simulation Optimization Techniques. Proceedings of the 1998 Winter Simulation Conference.

April, J., Glover, F., Kelly, J. P., Laguna, M. 2003. Practical Introduction to Simulation Optimization. Proceedings of the 2003 Winter Simulation Conference.

Banks, J., Carson, J. S., II, Nelson, B. L. 1996. Discrete-Event System Simulation. Prentice hall, New Jersey. 548 pp.

Barton, R. R. 1998. Simulation Metamodels. Proceedings of the 1998 Winter Simulation Conference.

Banks, J., Carson, J. S., II, Nelson, B. L., Nicol, D. M. 2001. Discrete-Event System Simulation. Prentice Hall, New Jersey. 594 pp.

Boesel, J., Glover, F., Bowden, R. O., Jr., Kelly, J. P., Westwig, E. 2001. Proceedings of the 2001 Winter Simulation Conference.

Charnes, J. M. 1991. Multivariate Simulation Output Analysis. Proceedings of the 1991 Winter Simulation Conference.

Chen, E. J. 2001. Using Common Random Numbers for Indifference-Zone Selection. Proceedings of the 2001 Winter Simulation Conference.

Chick, S. E. 2000. Bayesian Methods for Simulation. Proceedings of the 2000 Winter Simulation Conference.

Devore, J. L. 1982. Probability & Statistics for Engineering and the Sciences. Brooks/Cole Publishing Company, California, 640 pp.

dos Santos, M., I., R., Porta Nova, A. M. O. 1999. The Main Issues in Nonlinear Simulation Metamodel Estimation. Proceedings of the 1999 Winter Simulation Conference.

Fu, M. C. 2001. Simulation Optimization. Proceedings of the 2001 Winter Simulation Conference.

Goldsman, D., Nelson, B. L. 1998. Statistical Screening, Selection, and Multiple Comparison Procedures in Computer Simulation. Proceedings of the 1998 Winter Simulation Conference.

Ivanova, T., Malone, L. 1999. Comparison of a Two-Stage Group-Screening Design to a Standard 2^{k-p} Design for a Whole-Line Semiconductor Manufacturing Simulation Model. Proceedings of the 1999 Winter Simulation Conference.

Jacobson, S. H., Swisher, J. R. 1999. A Survey of Ranking, Selection, and Multiple Comparison Procedures for Discrete-Event Simulation. Proceedings of the 1999 Winter Simulation Conference.

Jin, J., Schmeiser, B. 2003. Simulation-Based Retrospective Optimization of Stochastic Systems: A Family of Algorithms. Proceedings of the 2003 Winter Simulation Conference.

Kelton, W. D. 1997. Statistical Analysis of Simulation Output. Proceedings of the 1997 Winter Simulation Conference.

Kelton, W. D., Sadowski, R. P., Sadowski, D. A. 1998. Simulation with Arena. WCB/McGraw-Hill, U. S. A., 547 pp.

Kelton, W. D., Barton, R. R. 2003. Experimental Design for Simulation. Proceedings of the 2003 Winter Simulation Conference.

Kleijnen, J. P. C., Noordegraaf, A.V., Nielen, M. 2001. Sensitivity Analysis of Censored Output Through Polynomial, Logistic, and Tobit Regression Meta-models: Theory and Case Study. Proceedings of the 2001 Winter Simulation Conference.

Law, A. M., Kelton, W. D. 1982. Simulation Modeling and Analysis. McGraw-Hill Book Company, U. S. A., 400 pp.

Law, A. M., Kelton, W. D. 1991. Simulation Modeling and Analysis. McGraw- Hill, Inc., New York, 758 pp.

Law, A. M. 2003. How to Conduct a Successful Simulation Study. Proceedings of the 2003 Winter Simulation Conference.

Martinich, J. S. 1997. Production and Operations Management: An Applied Modern Approach. John Wiley and Sons, Inc., Missouri-St. Louis, p. 782.

Montgomery, D. C. 2001. Design and Analysis of Experiment. John Wiley & Sons, Inc., New York, 684 pp.

Myers, R. H., Montgomery, D. C. 1995. Response Surface Methodology: Process and Product Optimization Using Designed Experiments. John Wiley & Sons, Inc., New York, p. 656-660.

Nelson, B. L., Kim, S-H. 2003a. Selecting the Best System: Theory and Methods. Proceedings of the 2003 Winter Simulation Conference.

Nelson, B. L., Hong, L. J. 2003b. An Indifference-Zone Selection Procedure with Minimum Switching and Sequential Sampling. Proceedings of the 2003 Winter Simulation Conference.

Pegden, C. D., Shannon, R. E., Sadowski, R. P. 1995. Introduction to Simulation Using SIMAN. McGraw-Hill, Inc., U. S. A., 600 pp.

Sanchez, S. 2001. ABC's of Output Analysis. Proceedings of the 2001 Winter Simulation Conference.

Sargent, R. G. 1998. Verification and Validation of Simulation Models. Proceedings of the 1998 Winter Simulation Conference.

Sargent, R. G., Nance, R. E. 2002. Perspectives on the Evolution of Simulation. Operations Research. Volume 50, Number 1: 161 – 172.

Stuckman, B., Evans, G., Mollaghasemi, M. 1991. Comparison of Global Search Methods for Design Optimization Using Simulation.

Swisher, J. R., Hyden, P. D., Jacobson, S. H., Schruben, L. W. 2000. A Survey of Simulation Optimization Techniques and Procedures. Proceedings of the 2000 Winter Simulation Conference.

Taha, H. A. 1995. "Investigaciones Operacionales". Alfaomega Grupo Editor, S.A. de C.V., Colombia, p. 2 -3.

Wan, H., Ankenman, B., Nelson, B. L. 2003. Controlled Sequential Bifurcation: A New Factor-Screening Method for Discrete-Event Simulation. Proceedings of the 2003 Winter Simulation Conference.

Welch, P. D.1981. On the Problem of the Initial Transient in Steady-State Simulation, IBM Watson Research Center, Yorkstown Heights, N.Y.

Welch, P. D. 1983. The Statistical Analysis of Simulation Results. In: S. S. Lavenberg (ed.). The Computer Performance Modeling Handbook, pp. 268 – 328.academic Press, New York.

Wilson, J. R., Lada, E. K., Steiger, N. M., Alexopoulus, C., Goldsman, D., Zouaoui, F. 2002. ASAP2: An Improved Batch Means Procedure for Simulation Output Analysis. Proceedings of the 2002 Winter Simulation Conference.

APPENDIX A. Summary Previous Project

As mentioned in section 1.1, this project is a continuation of a previous project that was developed as part of the requirements for a computer simulation course towards the Master degree in Engineering. This section is a summary of the previous project and includes some elements of the conceptual model, data gathering and analysis, simulation language coding, verification and validation of the model, trial and error runs, and conclusions and recommendations. The simulation software used for coding the model and for performing the simulation statistical analysis was ARENA/SIMAN.

The definition of the problem, the description of the system, the objectives and the graphical display of the system were already discussed in Chapter I of this document.

A.1. Conceptual Model Formulation

The following are the assumptions of the model:

- The inter-arrival time follows an exponential distribution and the arrival rate will depend upon the restrictions on the up-time for the equipment, the personnel utilization and the work-in process inventory.
- The batch size is of one entity and the first entity is created at time zero.
- The buffers have no capacity since in the real system the units are moved to the floor if necessary.
- Reworking units is considered an external event since it is done in the assembly area which is not in the scope of this project.
- There are two shifts of eight hours each. All employees take the breaks at the same time. Lunch time is for 30 minutes and there are 10 minutes break during the morning and the afternoon.
- There is no capacity lost due to machine breakages or stoppages. Downtime is being indirectly considered with the maximum up-time specified for each machine.

- The testing personnel have been trained to perform all the functions of the cell.
- The final inspector has been trained to perform all of the operations of the final inspection area. The assembler has not been trained to perform the final inspection.
- The testing personnel are independent resources; they do not help or assist each other.

The following are the definitions of the elements, attributes and types of events of the simulation model:

- *Dynamic entity* – the new product being manufactured at the assembly line.
- *Static entities* – dielectric machine, testing machines #1, testing machine #2, draw out machine, testing personnel #1, testing personnel #2, assembly personnel, final inspector, packing personnel #1, packing personnel #2 and dummy operators used for animation purposes.
- *Attributes* – the arrival time of the products into the system.
- *External events* – the arrival of the product into the system and the rework.
- *Internal events* – completion of the service or operation at each working station (dielectric, testing, adjustment, inspection I, draw out, final inspection and packing).

The following is a list of the state variables of the model with the description of the state:

- State of the dielectric testing machine (0 for busy and 1 for not busy).
- State of the testing machine #1 (0 for busy and 1 for not busy).
- State of the testing machine #2 (0 for busy and 1 for not busy).
- State of the draw out machine (0 for busy and 1 for not busy).
- State of the testing personnel #1 (0 for busy and 1 for not busy).
- State of the testing personnel #2 (0 for busy and 1 for not busy).
- State of the assembly personnel (0 for busy and 1 for not busy).

- State of the final inspector (0 for busy and 1 for not busy).
- State of the packing personnel #1 (0 for busy and 1 for not busy).
- State of the packing personnel #2 (0 for busy and 1 for not busy).
- Buffer size for entities waiting at the dielectric station (0, 1, 2, ∞).
- Buffer size for entities waiting for available testing personnel to perform the dielectric test (0, 1).
- Buffer size for entities waiting at the testing station #1 (0, 1, 2, ∞).
- Buffer size for entities waiting at the testing station #2 (0, 1, 2, ∞).
- Buffer size for entities waiting for an available testing personnel to load the testing machine #1 (0, 1).
- Buffer size for entities waiting for an available testing personnel to load the testing machine #2 (0, 1).
- Buffer size for entities waiting for an available testing personnel to unload the testing machine #1 (0, 1).
- Buffer size for entities waiting for an available testing personnel to unload the testing machine #2 (0, 1).
- Buffer size for entities waiting for adjustment at testing station #1 (0, 1, 2, ∞).
- Buffer size for entities waiting for adjustment at testing station #2 (0, 1, 2, ∞).
- Buffer size for entities waiting for inspection I (0, 1, 2, ∞).
- Buffer size for entities waiting for the draw out test (0, 1, 2, ∞).
- Buffer size for entities waiting for the final inspection (0, 1, 2, ∞).
- Buffer size for entities waiting for the packing process (0, 1, 2, ∞).

A.2. Data Gathering and Analysis

The new product is considered a custom made product since the customer can select the desired configuration from a set of alternatives in which thousands of combinations can be done. This was the major challenge in regards to the data gathering and analysis since it was almost impossible and unpractical to simulate all possible combinations. A data record sheet that considered all sources of variation or characteristics of the product was then developed and provided to the employees in the

area. The method was to analyze the data first by not considering any source of variation, and then to filter the data using the different characteristics of the product in order to adjust to a particular probability distribution if needed. The objective was to determine the minimum number of different characteristics to be simulated. For our benefit, we were able to explain the variability of each process by adjusting to a theoretical probability distribution without having to filter by the different characteristics or source of variation.

A.3. Verification of the Simulation Model

The model was coded following the concept of sub-models and using counters at the beginning and at the end of each sub-model in order to facilitate the verification of the model. The element "TRACE"[14] was also used as well as the animation feature of ARENA as part of the verification process.

A.4. Validation of the Simulation Model

The simulation model was validated to have an assurance that its behavior was in accordance to the actual results of the real system. The following are the steps followed during the validation process:

- The steady state was determined.
- The minimum number of runs or replicates was also determined.
- The validation methods of continuity, degenerate and absurd conditions were followed.
- A validation run was then performed by setting the parameters of the model to the actual conditions of the real system. The results were compared to the data provided by the management of the company.

[14] TRACE - is a feature of the simulation language SIMAN used to verify in detail the movement of entities through the system in order to identify situations wherein the entity flow is incorrect and/or the function performed at a specific block is incorrect.

Finally, the results from the validation run and from each of the validation methods were discussed with the management in order to gain their confidence with the model.

A.5. Trial and Error Experimental Run

The experimental phase followed a trial and error approach in which the main objective was to increase the production while restrictions on the uptime of the testing machines, utilization of personnel and work-in process inventory were maintained within target. The method was to increase the arrival rate until any of the restrictions was not satisfied. The following is a summary of each experimental run:

- *Run #1* – This is the validation run. It was concluded that the yield have a major impact on the throughput and that a third shift was necessary.
- *Runs #2 and #3* – These runs were done changing the acceptance rate or yield, and by adding a third shift. The draw out area became the bottleneck as yield was increased. It was observed that the utilization of personnel between the testing and the draw out areas was not balanced.
- *Run #4* – This run was done balancing the work load by moving out the Inspection I from the draw out area into the testing area, and by setting up the yield to 80%. A throughput of 1500 units per month was achieved for the first time. It was also observed that the up-time for the dielectric test was low, suggesting no need for investment in this equipment.
- *Runs #5 and #6* – The parameters were fixed to the same conditions of run #4 with the exception of the acceptance rate which was increased to 90% and 95%. Throughput increased by 200 units as compared to run #4 and it was concluded that only one packer was needed to achieve 1500 units per month.

A.6. Conclusions and Recommendations

It was concluded that the yield was a critical parameter and that improving the acceptance rate was necessary in order to minimize the investment. It was also concluded that a third shift was necessary as well as balancing the work load between the testing

area and the draw out area. Based on the 1500 units per month achieved during the experimental runs, the following recommendations were given in order to achieve 3000 units:

- *Testing area* – 1 dielectric test machine, 4 testing machines and 4 testing personnel.
- *Draw out area* – 2 draw out testing machines, 2 assembly operators and 2 final inspectors.
- *Packing area* – 2 packing personnel.

APPENDIX B. SIMAN Codes

Model:

```
0$      CREATE,1:expo(14.3,10):MARK(tiempodellegada):NEXT(33$);      Arrival of units into the system.

33$     COUNT:breakerstotalesgenerados,1;                            Counts units that arrived into the
system.
1$      ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema+1; Increase WIP by one unit.
133$    ASSIGN:picture=producto;                                     For animation purposes.

70$     STATION,entrada;                                             For animation purposes.
4$      ROUTE: .1,estaciondielectrico;                               Unit is moved to the dielectric test
area.


5$      STATION,estaciondielectrico;                                 Dielectric station
2$      QUEUE,filadielectrico;                                       Buffer for dielectric test.
91$     SEIZE,1,Other:maquinadielectrico,1:NEXT(92$);                Seizes dielectric machine.

92$     QUEUE,filadielectricodisponible;                             Waiting time when dielectric machine
is available but testing personnel is not.
90$     SELECT,CYC:3$:93$;                                           Select the testing personnel who is
available.
3$      SEIZE,3,Other:tester1,1:NEXT(136$);                          Seizes testing personnel #1.

136$    SEIZE,1,Other:testerdielectdummy1,1:NEXT(6$);                For animation purposes

6$      DELAY:1.06+logn(.423,.266,1),,Other:NEXT(8$);                Dielectric process time.

8$      RELEASE:maquinadielectrico,1;                                Releases dielectric machine.
73$     COUNT:#bkrspordielectrico,1;                                 Counts the number of units that tested
at the dielectric area.
7$      BRANCH,1,yielddielectric:With,.99,11$,Yes:Else,9$,Yes;       Simulates the rejection/acceptance of
units at the dielectric test.
11$     ROUTE:.1,estaciontesting1;                                   Accepted units are moved to the
testing area.

9$      COUNT:rechazadasendielectrico,1;                             Counts the number of units rejected at
the dielectric area.
34$     ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; Discounts WIP by one unit.
156$    RELEASE:testerdielectdummy1,1:tester1,1;                     Releases testing personnel.
10$     DISPOSE:No;

93$     SEIZE,3,Other:tester2,1:NEXT(137$);                          Seizes the testing personnel #2.

137$    SEIZE,1,Other:testerdielectdummy2,1:NEXT(94$);               For animation purposes.

94$     DELAY:1.06+logn(.423,.266,1),,Other:NEXT(96$);               Dielectric process time.

96$     RELEASE:maquinadielectrico,1;                                Releases dielectric machine.
101$    COUNT:#bkrspordielectrico,1;                                 Counts the number of units that tested
at the dielectric area.
95$     BRANCH,1,yielddielectric:With,.99,99$,Yes:Else,97$,Yes;      Simulates the rejection/acceptance of
units at the dielectric test.
99$     ROUTE:.1,estaciontesting2;                                   Accepted units are moved to the
testing area.
```

```
97$       COUNT:rechazadasendielectrico,1;                                  Counts the number of units rejected
at the dielectric area.
100$      ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; Discounts WIP by one unit.
157$      RELEASE:testerdielectdummy2,1:tester2,1;                          Releases testing personnel.
98$       DISPOSE:No;


14$       STATION,estaciontesting2;
153$      RELEASE:testerdielectdummy2,1:tester2,1;                          Releases of testing personnel once
the unit has been moved to the testing area.
12$       QUEUE,filatesting2;                                               Buffer for testing machine #2.
13$       SEIZE,1,Other:maquinatest2,1:NEXT(89$);                           Seizes testing machine #2 if
available.

89$       QUEUE,filamaqtestdisponible2;                                     Waiting time if testing personnel is
not available.
88$       SEIZE,1,Other:tester2,1:NEXT(141$);                               Seizes testing personnel if available.

141$      SEIZE,1,Other:testertestdummy2,1:NEXT(15$);                       For animation purposes.

15$       DELAY:1+logn(0.334,0.231,2),,Other:NEXT(142$);                    Loading time for the testing machine.

142$      RELEASE:testertestdummy2,1;                                       For animation purposes.
17$       RELEASE:tester2,1;                                                Releases testing personnel.
74$       COUNT:#bkrsporcargatest2,1;                                       Counts how many units were loaded
into the testing machine.
21$       DELAY:3+52*beta(2.93,4.65,3),,Other:NEXT(75$);                    Testing process time.

75$       COUNT: #bkrsportest2,1;                                           Counts how many units were tested.
22$       QUEUE,filabloqueotest2;                                           Waiting time when the testing
machine can not be unloaded because there is no testing personnel available.
23$       SEIZE,1,Other:tester2,1:NEXT(143$);                               Seizes testing personnel if available.

143$      SEIZE,1,Other:testertestdummy2,1:NEXT(24$);                       For animation purposes.

24$       DELAY:1.05+weib(.17,2.03,4),,Other:NEXT(25$);                     Unloading process time of the
testing machine.

25$       RELEASE:maquinatest2,1;                                           Releases testing machine.
76$       COUNT:#bkrspordescargatest2,1;                                    Counts how many units were
unloaded from the testing machine.
16$       BRANCH,1,yieldtesting:With,.726,20$,Yes:Else,18$,Yes;             Accepts/rejects units at the testing
station.
20$       ROUTE:.1,estacionajuste2;                                         Units are moved to the adjustment
area.

18$       COUNT:rechazadasentesting2,1;                                     Counts how many units were
rejected at testing.
36$       ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1;   Discounts WIP by one unit.
159$      RELEASE:testertestdummy2,1:tester2,1;                             Releases testing personnel.
19$       DISPOSE:No;

28$       STATION,estacionajuste2;
155$      RELEASE:testertestdummy2,1:tester2,1;                             Releases the testing personnel
once the units have been moved to the adjustment area.
26$       QUEUE,filaajuste2;                                                Buffer for the adjusting process.
27$       SEIZE,2,Other:tester2,1:NEXT(135$);                               Seizes testing personnel.

135$      SEIZE,1,Other:ajustedummy2,1:NEXT(29$);                           For animation purposes.
```

```
29$       DELAY:1.03+logn(0.369,0.202,5),,Other:NEXT(165$);          Adjusting process time.

165$      ASSIGN:test=2;                                             Attribute to identify the testing
personnel doing the job for a particular unit.
77$       COUNT:#bkrsporajuste2,1;
30$       BRANCH,1,yieldajuste:With,1.0,69$,Yes:Else,31$,Yes;        Accepts/rejects units at the
adjustment process.
69$       COUNT:#breakersportest2,1;                                 Counts how many units passed the
adjustment process.
47$       ROUTE:.1,estacioninspeccion1;                              Units are moved to the inspection
area.

31$       COUNT:rechazadasenajuste2,1;                               Counts how many units were rejected
at the adjustment process.
35$       ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; Decreases WIP by one unit.
166$      RELEASE:ajustedummy2,1:tester2,1;                          Releases testing personnel.
32$       DISPOSE:No;


39$       STATION,estacioninspeccion1;
162$      BRANCH,1:If,test.eq.1,163$,Yes:Else,164$,Yes;              Decides on releasing the appropriate
testing personnel.
163$      RELEASE:ajustedummy1,1:tester1,1;                          Releases testing personnel #1.
37$       QUEUE,filainspeccion1;                                     Buffer for inspection I.
45$       SELECT,POR:38$:46$;                                        Decides whether to seize the
assembler or the final inspector.
38$       SEIZE,2,Other:assembler,1:NEXT(167$);                      Seizes the assembler, which ha a
higher priority.

167$      ASSIGN:test=1;                                             Attribute to identify the personnel
selected.
144$      SEIZE,1,Other:assemdummyinsp1,1:NEXT(40$);                 For animation purposes.

40$       DELAY:tria(5.5,10.5,18.5,6),,Other:NEXT(72$);              Inspection I process time.

72$       COUNT:#bkrsporinsp1,1;                                     Counts how many units were
inspected.
41$       BRANCH,1,yieldinspectionI:With,.975,44$,Yes:Else,42$,Yes;  Accepts/Rejects units at the inspection
I area.
44$       ROUTE:.1,estaciondrawout;                                  Units are moved to the draw out area.

42$       COUNT:rechazadaseninsp1,1;                                 Counts how many units were rejected
at the draw out area.
60$       ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1;Decreases WIP by  one unit.
168$      BRANCH,1:If,test.eq.1,169$,Yes:Else,170$,Yes;              defines who did the inspection I for a
particular unit.
169$      RELEASE:assemdummyinsp1,1:assembler,1;                     Releases assembler.
43$       DISPOSE:No;

170$      RELEASE:fininspdummyinsp1,1:finalinspector,1:NEXT(43$);    Releases final inspector.

46$       SEIZE,3,Other:finalinspector,1:NEXT(178$);                 Seizes the final inspector, which ha a
lower priority.

178$      ASSIGN:test=2;                                             Attribute to identify the personnel
selected.
145$      SEIZE,1,Other:fininspdummyinsp1,1:NEXT(48$);               For animation purposes.

48$       DELAY:tria(5.5,10.5,18.5,6),,Other:NEXT(72$);              Inspection I process time.
```

```
164$    RELEASE:ajustedummy2,1:tester2,1:NEXT(37$);          Releases testing personnel #2.


50$     STATION,estaciondrawout;
171$    BRANCH,1:If,test.eq.1,172$,Yes:Else,173$,Yes;        Selects the personnel who did the
inspection I.
172$    RELEASE:assemdummyinsp1,1:assembler,1;               Releases the assembler.
49$     QUEUE,filadrawout;                                   Buffer for draw out test.
79$     SEIZE,2,Other:maquinadrawout,1:NEXT(80$);            Seizes draw out machine.

80$     QUEUE,filadrawoutdisponible;                         Waiting time if the assembler or final
inspector are not available.
59$     SELECT,POR:54$:57$;                                  Defines who will perform the draw out
test.
54$     SEIZE,1,Other:assembler,1:NEXT(179$);                Seizes the assembler.

179$    ASSIGN:test=1;                                       Identify the personnel who was
selected to perform the draw out test.
146$    SEIZE,1,Other:assemdummydwout1,1:NEXT(55$);          For animation purposes.

55$     DELAY:7.5+weib(8.6,2.32,7),,Other:NEXT(186$);        Draw out processing time.

186$    RELEASE:maquinadrawout,1;                            Releases the draw out machine.
71$     COUNT:#bkrspordrawout,1;                             Counts how many units passed through
draw out.
51$     BRANCH,1,yielddrawout:With,.961,53$,Yes:Else,52$,Yes;  Accepts/rejects units at the draw out
area.
53$     ROUTE:.1,estacioninspeccionfinal;                    Units are moved to final inspection.

52$     COUNT:rechazadasendrawout,1;                         Counts how many units were rejected
at the draw out area.
56$     ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; Decreases WIP by one unit.
175$    BRANCH,1:If,test.eq.1,176$,Yes:Else,177$,Yes;        Selects the employee who performed
the draw out test for a particular unit.
176$    RELEASE:assemdummydwout1,1:assembler,1;              Releases the assembler.
174$    DISPOSE:No;

177$    RELEASE:fininspdummydwout1,1:finalinspector,1:NEXT(174$);  Releases the final inspector.

57$     SEIZE,2,Other:finalinspector,1:NEXT(180$);           Seizes the final inspector.

180$    ASSIGN:test=2;                                       Identify the personnel who was
selected to perform the draw out test.
147$    SEIZE,1,Other:fininspdummydwout1,1:NEXT(58$);        For animation purposes.

58$     DELAY:7.5+weib(8.6,2.32,7),,Other:NEXT(186$);        Draw out processing time.

173$    RELEASE:fininspdummyinsp1,1:finalinspector,1:NEXT(49$);  Releases the final inspector.

62$     STATION,estacioninspeccionfinal;
181$    BRANCH,1:If,test.eq.1,182$,Yes:Else,183$,Yes;        Selects whoever performed the draw
out test for a particular unit.
182$    RELEASE:assemdummydwout1,1:assembler,1;              Releases the assembler.
61$     QUEUE,filainspeccionfinal;                           Buffer for final inspection.
66$     SEIZE,1,Other:finalinspector,1:NEXT(148$);           Seizes the final inspector.

148$    SEIZE,1,Other:fininspdummyinspfin1,1:NEXT(67$);      For animation purposes.

67$     DELAY:3.5+expo(3.27,8),,Other:NEXT(78$);             Final inspection processing time.
```

```
78$         COUNT:#bkrsporinspfinal,1;                                  Counts how many units passed through
final inspection.
63$         BRANCH,1,yieldfinalinspection:With,.948,86$,Yes:Else,64$,Yes;   Accepts/Inspects units at final
inspection area.
86$         ROUTE:.1,estacionempaque;                                   Units are moved to packing.

64$         COUNT:rechazadaseninspfinal,1;                              counts how many units were rejected at
the final inspection area.
68$         ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; Decreases WIP by one unit.
184$        RELEASE:fininspdummyinspfin1,1:finalinspector,1;            Releases the final inspector.
65$         DISPOSE:No;

183$        RELEASE:fininspdummydwout1,1:finalinspector,1:NEXT(61$);    Releases the final inspector.

81$         STATION,estacionempaque;
185$        RELEASE:fininspdummyinspfin1,1:finalinspector,1:NEXT(149$);   Releases final inspector once the unit
is moved into the packing area.


;
;
;    Model statements for module:  Seize 1
;
149$        QUEUE,filaempaque;
187$        SEIZE,1,Other:SELECT(empaque,CYC,),1:NEXT(192$);

192$        ASSIGN:j=j;
189$        DELAY:0.000,,Other:NEXT(82$);

82$         DELAY:8+9*beta(1.15,.554,9),,Other:NEXT(151$);              Packing processing time.


;
;
;    Model statements for module:  Release 1
;
151$        TRACE,-1,"-Releasing resources\n":;
193$        RELEASE:SELECT(empaque,LAST),1:NEXT(83$);

83$         ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; decreases WIP by one unit.
84$         TALLY:tiempoenelsistema,tnow-tiempodellegada,1;             Calculates the cycle time for each unit.
87$         COUNT:bkrsembarcados,1;                                     Counts How many units were packed
(Throughput)
85$         DISPOSE:No;


104$        STATION,estaciontesting1;
152$        RELEASE:testerdielectdummy1,1:tester1,1;                    Releases of testing personnel once the
unit has been moved to the testing area.
102$        QUEUE,filatesting1;                                         Buffer for testing machine #1.
103$        SEIZE,1,Other:maquinatest1,1:NEXT(121$);                    Seizes testing machine #1 if available.

121$        QUEUE,filamaqtestdisponible1;                               Waiting time if testing personnel is not
available.
120$        SEIZE,1,Other:tester1,1:NEXT(138$);                         Seizes testing personnel if available.

138$        SEIZE,1,Other:testertestdummy1,1:NEXT(105$);                For animation purposes.

105$        DELAY:1+logn(0.334,0.231,2),,Other:NEXT(139$);              Loading time for the testing machine.
```

```
139$      RELEASE:testertestdummy1,1;                                  For animation purposes.
107$      RELEASE:tester1,1;                                           Releases testing personnel.
117$      COUNT:#bkrsporcargatest1,1;                                  Counts how many units were loaded
into the testing machine.
111$      DELAY:3+52*beta(2.93,4.65,3),,Other:NEXT(118$);              Testing process time.

118$      COUNT:#bkrsportest1,1;                                       Counts how many units were tested.
112$      QUEUE,filabloqueotest1;                                      Waiting time when the testing machine
can not be unloaded because there is no testing personnel available.
113$      SEIZE,1,Other:tester1,1:NEXT(140$);                          Seizes testing personnel if available.

140$      SEIZE,1,Other:testertestdummy1,1:NEXT(114$);                 For animation purposes.

114$      DELAY:1.05+weib(.17,2.03,4),,Other:NEXT(115$);               Unloading process time of the testing
machine.

115$      RELEASE:maquinatest1,1;                                      Releases testing machine.
119$      COUNT:#bkrspordescargatest1,1;                               Counts how many units were unloaded
from the testing machine.
106$      BRANCH,1,yieldtesting:With,.726,110$,Yes:Else,108$,Yes;      Accepts/rejects units at the testing
station.
110$      ROUTE:.1,estacionajuste1;                                    Units are moved to the adjustment area.

108$      COUNT:rechazadasentesting1,1;                                Counts how many units were rejected
at testing.
116$      ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; Discounts WIP by one unit.
158$      RELEASE:testertestdummy1,1:tester1,1;                        Releases testing personnel.
109$      DISPOSE:No;

124$      STATION,estacionajuste1;
154$      RELEASE:testertestdummy1,1:tester1,1;                        Releases the testing personnel once the
units have been moved to the adjustment area.
122$      QUEUE,filaajuste1;                                           Buffer for the adjusting process.
123$      SEIZE,2,Other:tester1,1:NEXT(134$);                          Seizes testing personnel.

134$      SEIZE,1,Other:ajustedummy1,1:NEXT(125$);                     For animation purposes.

125$      DELAY:1.03+logn(0.369,0.202,5),,Other:NEXT(160$);            Adjusting process time.

160$      ASSIGN:test=1;                                               Attribute to identify the testing
personnel doing the job for a particular unit.
132$      COUNT:#bkrsporajuste1,1;
126$      BRANCH,1,yieldajuste:With,1.0,131$,Yes:Else,127$,Yes;        Accepts/rejects units at the adjustment
process.
131$      COUNT:#breakersportest1,1;                                   Counts how many units passed the
adjustment process.
130$      ROUTE:.1,estacioninspeccion1;                                Units are moved to the inspection area.

127$      COUNT:rechazadasenajuste1,1;                                 Counts how many units were rejected
at the adjustment process.
129$      ASSIGN:#breakerspromedioensistema=#breakerspromedioensistema-1; Decreases WIP by one unit.
161$      RELEASE:ajustedummy1,1:tester1,1;                            Releases testing personnel.
128$      DISPOSE:No;
```

Experiment:

```
PROJECT,     "Actual System - Validation Run","Roberto L. Seijo Vidal",03/11/2004,No,No,No,No,No,No,No,No;

ATTRIBUTES:  test:
```

```
        tiempodellegada;

VARIABLES:    #breakerspromedioensistema,CLEAR(System),CATEGORY("None-None");

SEEDS:        yieldinspectionI,456789,No:
        yielddielectric,123456,No:
        yieldfinalinspection,678912,No:
        yieldajuste,345678,No:
        yieldtesting,234567,No:
        yielddrawout,567891,No;

QUEUES:       filamaqtestdisponible1,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filamaqtestdisponible2,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filadrawout,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filatesting1,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filatesting2,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filadielectricodisponible,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filadrawoutdisponible,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filainspeccionfinal,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filabloqueotest1,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filabloqueotest2,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filadielectrico,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filainspeccion1,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filaempaque,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filaajuste1,FirstInFirstOut,,AUTOSTATS(Yes,,):
        filaajuste2,FirstInFirstOut,,AUTOSTATS(Yes,,);

PICTURES:     producto;

RESOURCES:    testerdielectdummy1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        testerdielectdummy2,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        maquinatest1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        maquinatest2,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        maquinadrawout,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        maquinadielectrico,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        fininspdummydwout1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        assembler,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        assemdummyinsp1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        ajustedummy1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        ajustedummy2,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        tester1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        tester2,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        fininspdummyinsp1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        fininspdummyinspfin1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        testertestdummy1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        testertestdummy2,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        finalinspector,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        assemdummydwout1,Capacity(1),,Stationary,COST(0.0,0.0,0.0),,AUTOSTATS(Yes,,):
        empacador1:
        empacador2;

STATIONS:     estacioninspeccion1:
        estacioninspeccionfinal:
        entrada:
        estacionajuste1:
        estacionajuste2:
        estacionempaque:
        estaciondrawout:
        estaciondielectrico:
        estaciontesting1:
```

```
        estaciontesting2;

COUNTERS:   1,breakerstotalesgenerados,,Replicate:
        2,#bkrspordielectrico,,Replicate:
        3,#bkrsportest1,,Replicate:
        4,#bkrsporcargatest1,,Replicate:
        5,#bkrspordescargatest1,,Replicate:
        6,#breakersportest1,,Replicate:
        7,#bkrsporajuste1,,Replicate:
        8,#bkrsportest2,,Replicate:
        9,#bkrsporcargatest2,,Replicate:
        10,#bkrspordescargatest2,,Replicate:
        11,#breakersportest2,,Replicate:
        12,#bkrsporajuste2,,Replicate:
        13,#bkrsporinsp1,,Replicate:
        14,#bkrspordrawout,,Replicate:
        15,#bkrsporinspfinal,,Replicate:
        16,bkrsembarcados,,Replicate:
        17,rechazadasendielectrico,,Replicate:
        18,rechazadasentesting1,,Replicate:
        19,rechazadasenajuste1,,Replicate:
        20,rechazadasentesting2,,Replicate:
        21,rechazadasenajuste2,,Replicate:
        22,rechazadaseninsp1,,Replicate:
        23,rechazadasendrawout,,Replicate:
        24,rechazadaseninspfinal,,Replicate;

TALLIES:    tiempoenelsistema,"tsisindi1.dat";

DSTATS:     1,#breakerspromedioensistema,Avg.Units in System:
        2,nq(filadielectrico),Fila Dielectrico:
        3,nq(filadielectricodisponible),Fila Dielectrico.Disp.:
        4,nq(filatesting1),Fila Testing1:
        5,nq(filamaqtestdisponible1),Fila Maq.Test1 Disp.:
        6,nq(filabloqueotest1),Fila Maq.Test1 Bloq.:
        7,nq(filaajuste1),Fila Ajuste1:
        8,nq(filatesting2),Fila Testing2:
        9,nq(filamaqtestdisponible2),Fila Maq.Test2 Disp.:
        10,nq(filabloqueotest2),Fila Maq.Test2 Bloq.:
        11,nq(filaajuste2),Fila Ajuste2:
        12,nq(filainspeccion1),Fila Insp.I:
        13,nq(filadrawout),Fila D.O.:
        14,nq(filadrawoutdisponible),Fila D.O.Disp.:
        15,nq(filainspeccionfinal),Fila Insp.Final:
        16,NQ(filaempaque),Fila Empaque:
        17,nr(maquinadielectrico),% Util.Maq.Dielectrico:
        18,nr(maquinatest1),% Util.Maq.Test1:
        19,nr(maquinatest2),% Util.Maq.Test2:
        20,nr(tester1),% Util.Tester1:
        21,nr(tester2),% Util.Tester2:
        22,nr(maquinadrawout),% Util.Maq.D.O.:
        23,nr(assembler),% Util.Assembler:
        24,nr(finalinspector),% Util.Final Insp.:
        25,(nr(empacador1)+nr(empacador2))/2,% Util. Packers;

OUTPUTS:    1,nc(breakerstotalesgenerados):
        2,nc(#bkrspordielectrico):
        3,nc(#bkrsportest1):
        4,nc(#bkrsporcargatest1):
        5,nc(#breakersportest1):
```

```
        6,nc(#bkrspordescargatest1):
        7,nc(#bkrsporajuste1):
        8,nc(#bkrsportest2):
        9,nc(#bkrsporcargatest2):
        10,nc(#breakersportest2):
        11,nc(#bkrspordescargatest2):
        12,nc(#bkrsporajuste2):
        13,nc(rechazadasendielectrico):
        14,nc(rechazadasentesting1):
        15,nc(rechazadasenajuste1):
        16,nc(rechazadasentesting2):
        17,nc(rechazadasenajuste2):
        18,davg(% Util.Maq.Dielectrico),"Up-time Dielectric machine 1.dat":
        19,davg(% Util.Maq.Test1),"%Up-time Tester machine #1 1.dat":
        20,davg(% Util.Maq.Test2),"%Up-time Tester machine #2 1.dat":
        21,davg(% Util.Tester1),"%Util.testing personnel #1 1.dat":
        22,davg(% Util.Tester2),"%Util.testing personnel #2 1.dat":
        23,davg(Fila Dielectrico):
        24,davg(Fila Dielectrico.Disp.):
        25,davg(Fila Testing1):
        26,davg(Fila Maq.Test1 Disp.):
        27,davg(Fila Maq.Test1 Bloq.):
        28,davg(Fila Ajuste1):
        29,davg(Fila Testing2):
        30,davg(Fila Maq.Test2 Disp.):
        31,davg(Fila Maq.Test2 Bloq.):
        32,davg(Fila Ajuste2):
        33,nc(#bkrsporinsp1):
        34,nc(#bkrspordrawout):
        35,nc(#bkrsporinspfinal):
        36,nc(rechazadaseninsp1):
        37,nc(rechazadasendrawout):
        38,nc(rechazadaseninspfinal):
        39,davg(% Util.Maq.D.O.),"%Up-time Draw Out machine 1.dat":
        40,davg(% Util.Assembler),"%Util.assembler 1.dat":
        41,davg(% Util.Final Insp.),"%Util.Final Inspector 1.dat":
        42,davg(Fila Insp.I):
        43,davg(Fila D.O.):
        44,davg(Fila D.O.Disp.):
        45,davg(Fila Insp.Final):
        46,nc(bkrsembarcados),"Throughput1.dat":
        47,davg(% Util. Packers),"%Util.packers 1.dat":
        48,davg(Fila Empaque):
        49,davg(Avg.Units in System),"WIP1.dat":
        50,tavg(tiempoenelsistema),"Avg.processing time1.dat";

REPLICATE,    100,0.0,19320,Yes,Yes,920,,,24.0,Minutes,No,No;

SETS:         empaque,empacador1,empacador2;
```

APPENDIX C. Verification warm-up period – Input parameters

Actual System vs. Warm-up Validation

Actual System: Warm-up after 920 minutes
Warm-Up Validation: Warm-up after 9,200 minutes

Simulation Parameters	Testing Area	Actual System	Warm-up Validation
	Inter-arrival time (Mins.)	Expo (14.3)	Expo (14.3)
	Qty. Dielectric testing machines	1	1
	Qty. testing personnel - Tester #1	1	1
	Qty. testing personnel - Tester #2	1	1
	Qty. testing machines - Tester #1	1	1
	Qty. testing machines - Tester #2	1	1
	Tester#1 processing time	3+52*Beta(2.93,4.65)	3+52*Beta(2.93,4.65)
	Tester #2 processing time	3+52*Beta(2.93,4.65)	3+52*Beta(2.93,4.65)
	Yield % - Tester #1	72.6	72.6
	Yield % - Tester #2	72.6	72.6
	Draw Out Area	**Actual System**	**Warm-up Validation**
	Qty. Assy. personnel	1	1
	Qty. Final Insp. personnel	1	1
	Qty. Draw Out testing machines	1	1
	Packing Area	**Actual System**	**Warm-up Validation**
	Qty. packing personnel	2	2

APPENDIX D. 2^{8-2} Fractional Factorial Design with Center Point

———— 3/25/2004 11:50:00 AM ————————————————

* NOTE * The number of centerpoints specified is doubled for each categorical factor. For Q categorical factors, the result is 2**Q times as many centerpoints.

Results for: Worksheet 7

Fractional Factorial Design

Factors:	8	Base Design:	8, 64	Resolution:	V
Runs:	74	Replicates:	1	Fraction:	1/4
Blocks:	1	Center pts (total):	10		

Design Generators: G = ABCD, H = ABEF

Alias Structure (up to order 4)

I

A + BCDG + BEFH
B + ACDG + AEFH
C + ABDG
D + ABCG
E + ABFH
F + ABEH
G + ABCD
H + ABEF
AB + CDG + EFH
AC + BDG
AD + BCG
AE + BFH
AF + BEH
AG + BCD
AH + BEF
BC + ADG
BD + ACG
BE + AFH
BF + AEH
BG + ACD
BH + AEF
CD + ABG + EFGH
CE + DFGH
CF + DEGH
CG + ABD + DEFH
CH + DEFG
DE + CFGH
DF + CEGH
DG + ABC + CEFH
DH + CEFG
EF + ABH + CDGH
EG + CDFH

```
EH + ABF + CDFG
FG + CDEH
FH + ABE + CDEG
GH + CDEF
ACE + BCFH + BDEG
ACF + BCEH + BDFG
ACH + BCEF + BDGH
ADE + BCEG + BDFH
ADF + BCFG + BDEH
ADH + BCGH + BDEF
AEG + BCDE + BFGH
AFG + BCDF + BEGH
AGH + BCDH + BEFG
BCE + ACFH + ADEG
BCF + ACEH + ADFG
BCH + ACEF + ADGH
BDE + ACEG + ADFH
BDF + ACFG + ADEH
BDH + ACGH + ADEF
BEG + ACDE + AFGH
BFG + ACDF + AEGH
BGH + ACDH + AEFG
CDE + FGH + ABEG
CDF + EGH + ABFG
CDH + EFG + ABGH
CEF + DGH + ABCH
CEG + DFH + ABDE
CEH + DFG + ABCF
CFG + DEH + ABDF
CFH + DEG + ABCE
CGH + DEF + ABDH
```

Design Table (randomized)

Run	A	B	C	D	E	F	G	H
1	-	+	+	+	-	-	-	-
2	+	+	+	-	+	+	-	+
3	-	-	+	-	+	-	-	-
4	-	-	-	-	+	-	+	-
5	+	+	+	+	-	-	+	+
6	-	+	+	+	-	+	-	+
7	+	+	-	+	-	+	-	-
8	+	+	-	+	+	+	-	+
9	+	-	-	+	+	-	+	+
10	+	-	+	+	-	+	-	+
11	-	+	+	-	-	+	+	+
12	-	-	+	-	+	+	-	+
13	-	+	+	+	+	+	-	-
14	0	0	0	0	0	0	0	+
15	-	-	+	+	+	+	+	+
16	-	+	+	+	+	-	-	+
17	-	+	-	-	-	-	-	-
18	-	-	+	-	-	+	-	-
19	-	+	+	-	+	+	+	-
20	0	0	0	0	0	0	0	-
21	-	+	-	+	+	-	+	+
22	+	+	-	-	+	-	+	-
23	+	+	+	+	+	-	+	-
24	+	+	-	-	-	+	+	-
25	+	-	+	-	+	-	+	+

26	+	+	−	+	−	−	−	+
27	0	0	0	0	0	0	0	+
28	+	+	+	+	−	+	+	−
29	+	+	−	−	−	−	+	+
30	−	+	−	+	+	+	+	−
31	−	−	−	−	−	+	+	−
32	0	0	0	0	0	0	0	−
33	+	−	+	−	+	+	+	−
34	+	+	−	−	+	+	+	+
35	+	+	+	−	−	−	−	+
36	0	0	0	0	0	0	0	+
37	+	−	+	−	−	+	+	+
38	0	0	0	0	0	0	0	−
39	+	−	+	+	+	−	−	+
40	−	−	+	+	−	−	+	+
41	0	0	0	0	0	0	0	−
42	0	0	0	0	0	0	0	−
43	−	+	−	−	−	+	−	+
44	+	−	−	−	+	−	−	+
45	−	−	−	+	+	−	−	−
46	−	+	−	+	−	−	+	−
47	−	−	−	+	−	−	−	+
48	−	+	+	−	−	−	+	−
49	+	−	+	+	−	−	−	−
50	+	+	+	+	+	+	+	+
51	+	−	−	+	−	+	+	+
52	+	−	−	−	−	+	−	+
53	+	+	−	+	+	−	−	−
54	−	−	−	−	−	−	+	+
55	0	0	0	0	0	0	0	+
56	0	0	0	0	0	0	0	+
57	−	−	−	−	+	+	+	+
58	+	−	+	−	−	−	+	−
59	−	+	−	−	+	+	−	−
60	−	+	−	−	+	−	−	+
61	+	−	−	+	+	+	+	−
62	−	−	−	+	−	+	−	−
63	+	−	−	+	−	−	+	−
64	−	+	−	+	−	+	+	+
65	+	−	−	−	−	−	−	−
66	−	−	+	+	+	−	+	−
67	+	−	+	+	+	+	−	−
68	+	+	+	−	−	+	−	−
69	+	−	−	−	+	+	−	−
70	−	−	−	+	+	+	−	+
71	−	+	+	−	+	−	+	+
72	+	+	+	−	+	−	−	−
73	−	−	+	+	−	+	+	−
74	−	−	+	−	−	−	−	+

APPENDIX E. 2^3 Factorial Design

Full Factorial Design

```
Factors:   3   Base Design:          3, 8
Runs:     24   Replicates:              3
Blocks:    3   Center pts (total):      0


Block Generators:  replicates


All terms are free from aliasing.



Design Table (randomized)

Run  Block  A  B  C
  1      2  -  +  +
  2      2  -  +  -
  3      2  -  -  -
  4      2  -  -  +
  5      2  +  -  +
  6      2  +  +  -
  7      2  +  +  +
  8      2  +  -  -
  9      3  -  -  +
 10      3  +  -  -
 11      3  -  +  +
 12      3  +  -  +
 13      3  -  +  -
 14      3  +  +  -
 15      3  +  +  +
 16      3  -  -  -
 17      1  -  +  +
 18      1  +  -  -
 19      1  +  +  +
 20      1  +  -  +
 21      1  -  +  -
 22      1  -  -  +
 23      1  -  -  -
 24      1  +  +  -
```

APPENDIX F. Central Composite Factorial Design

```
Welcome to Minitab, press F1 for help.
```

Central Composite Design

```
Factors:       3     Replicates:     2
Base runs:    20     Total runs:    40
Base blocks:   1     Total blocks:   2

Two-level factorial: Full factorial

Cube points:             16
Center points in cube:   12
Axial points:            12
Center points in axial:   0

Alpha: 1.68179
```

Design Table (randomized)

Run	Blk	A	B	C
1	2	0.00000	0.00000	0.00000
2	2	1.00000	1.00000	1.00000
3	2	-1.00000	1.00000	-1.00000
4	2	1.00000	-1.00000	1.00000
5	2	-1.00000	1.00000	1.00000
6	2	1.00000	-1.00000	-1.00000
7	2	1.68179	0.00000	0.00000
8	2	0.00000	0.00000	1.68179
9	2	0.00000	0.00000	0.00000
10	2	0.00000	1.68179	0.00000
11	2	0.00000	0.00000	0.00000
12	2	-1.00000	-1.00000	-1.00000
13	2	0.00000	-1.68179	0.00000
14	2	0.00000	0.00000	0.00000
15	2	0.00000	0.00000	0.00000
16	2	0.00000	0.00000	0.00000
17	2	1.00000	1.00000	-1.00000
18	2	0.00000	0.00000	-1.68179
19	2	-1.68179	0.00000	0.00000
20	2	-1.00000	-1.00000	1.00000
21	1	1.00000	-1.00000	1.00000
22	1	0.00000	0.00000	0.00000
23	1	-1.00000	-1.00000	-1.00000
24	1	1.68179	0.00000	0.00000
25	1	-1.00000	1.00000	-1.00000
26	1	0.00000	1.68179	0.00000
27	1	0.00000	0.00000	0.00000
28	1	-1.00000	1.00000	1.00000
29	1	0.00000	0.00000	1.68179
30	1	0.00000	-1.68179	0.00000
31	1	-1.00000	-1.00000	1.00000
32	1	0.00000	0.00000	0.00000
33	1	0.00000	0.00000	0.00000
34	1	1.00000	1.00000	-1.00000
35	1	1.00000	1.00000	1.00000

36	1	0.00000	0.00000	0.00000
37	1	0.00000	0.00000	0.00000
38	1	1.00000	-1.00000	-1.00000
39	1	0.00000	0.00000	-1.68179
40	1	-1.68179	0.00000	0.00000

APPENDIX G. Visual Basic Program – Optimization Tool

```
Option Explicit 'This is just for specifying that all variables must de defined


Private Sub cmbCases_Click()
'Purpose:
'This subroutine will allow us to select the appropriate case considering values of two
variables
'Variables

'Variable definition ---------------------------------------

'Variable Yield
Dim y As Double

'Variable Testing Process Time
Dim tpt As Double

'Upper and lower limits for Yield
Dim uly As Double
Dim lly As Double

'Upper and lower limits for Testing Process Time
Dim ultpt As Double
Dim lltpt As Double

'Variable that will evaluate user's choice
Dim choice As Integer

'Variable definition ends ----------------------------------


'Variables take value from cells
y = Cells(5, 2)
tpt = Cells(7, 2)
MsgBox "You have selected this values: y = " & y & "  and tpt =  " & tpt, vbInformation,
"Variables values"


'Bounds for variables are defined (for 3 testing machines)

'Yield
lly = 0.85
```

```
uly = 0.9

'Testing Process Time
lltpt = 0.7
ultpt = 0.75


'Compare the values of the variables to the range------------------
If y >= lly And y <= uly And tpt >= lltpt And tpt <= ultpt Then
   choice = 3
Else

   'Bounds for variables are defined (for 4 testing machines)

   'Yield
   lly = 0.8
   uly = 0.9

   'Testing Process Time
   lltpt = 0.7
   ultpt = 1

   'Compare the values of the variables to the range------------------

   If y >= lly And y <= uly And tpt >= lltpt And tpt <= ultpt Then
      choice = 4
   Else

      'Bounds for variables are defined (for 5 testing machines)

      'Yield
      lly = 0.726
      uly = 0.9

      'Testing Process Time
      lltpt = 0.7
      ultpt = 1

      'Compare the values of the variables to the range------------------
      If y >= lly And y <= uly And tpt >= lltpt And tpt <= ultpt Then
         choice = 5
      Else

         'Data entry error-----------------------------
         choice = 0
```

```
      End If
   End If
End If

'Compare the values of the variables to the range ENDS----------

Sheets("3 Testing Machines").Cells(5, 7) = y
Sheets("3 Testing Machines").Cells(7, 7) = tpt
Sheets("4 Testing Machines").Cells(5, 7) = y
Sheets("4 Testing Machines").Cells(7, 7) = tpt
Sheets("5 Testing Machines").Cells(5, 7) = y
Sheets("5 Testing Machines").Cells(7, 7) = tpt


'Selection of alternative of problem----------------------------------

If choice <> 0 Then
   Select Case choice

      Case 3
         Sheets("3 Testing Machines").Select
         MsgBox "You selected case for 3 Testing Machines", vbInformation, "Case
selected"

      Case 4
         Sheets("4 Testing Machines").Select
         MsgBox "You selected case for 4 Testing Machines", vbInformation, "Case
selected"

      Case 5
         Sheets("5 Testing Machines").Select
         MsgBox "You selected case for 5 Testing Machines", vbInformation, "Case
selected"

   End Select
Else
   MsgBox "You did not provide a valid value, try again", vbCritical, "You did not
provide a valid value"
End If

'Selection of alternative of problem ENDS---------------------------------


End Sub
```