A Heuristic for Multi Variable Goal Driven Simulation

Tuba Sevimoglu

FLORIDA INTERNATIONAL UNIVERSITY

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

A HEURISTIC FOR MULTI VARIABLE GOAL DRIVEN SIMULATION

A thesis submitted in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE

in

INDUSTRIAL ENGINEERING

by

Tuba Sevimoglu

2002

To: Dean Vish Prasad
College of Engineering

This thesis, written by Tuba Sevimoglu, and entitled A Heuristic for Multi Variable Goal Driven Simulation, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Joe Chow

Ronald Giachetti

Martha A. Centeno, Major Professor

Date of Defense: November 7, 2002

The thesis of Tuba Sevimoglu is approved.

Vish Prasad

Dean Vish Prasad
College of Engineering

Dean Douglas Wartzok
University Graduate School

Florida International University, 2002

DEDICATION

To my daughter Verda Nur, I love you very much.

A note: Verda, if you ever get to read this, I want you to know
that you were the first kid ever in the ARISE Center,
and that you were wonderful!

Dr. Centeno

October 22, 2002

ACKNOWLEDGMENTS

I wish to thank the members of my committee Dr. Ronald Giachetti and Dr. Joe Chow for their helpful comments.

A very special thanks must go to my major professor, Dr. Martha A. Centeno, for her support, unconditional patience and help, and specially for having the confidence in me to give me the chance to do this project. Most especially thank you for bringing in all the Disney® movies, and making my daughter addicted to them. ☺

Furthermore, I would like to thank Marsha A. Lee and Ingrid Lee for helping and encouraging me in completing this work.

I also wish to thank my daughter Verda Nur for letting me work on my thesis, and coming with me to the lab everyday, my mom Fatma Aral, my dad Necdet Aral, and my husband Orhan Sevimoglu for their support.

ABSTRACT OF THE THESIS

A HEURISTIC FOR MULTI VARIABLE GOAL DRIVEN SIMULATION

by

Tuba Sevimoglu

Florida International University, 2002

Miami, Florida

Professor Martha A. Centeno, Major Professor

This research sought to develop a heuristic for multi variable *on-line* GDS. Response Surface Methodology and Multi Attribute Utility Theory yielded first-degree polynomials for six measures of performance. The polynomials were used to develop 540 general decision rules to change input parameters (number of nurses, doctors, and beds). The multi-variable heuristic (MVH) utilizes these rules to automatically derive the simulation towards a desired performance goal. The MVH was embedded in an ER simulation model, using VBA for ARENA. The model served as the experimental tool to see if the MVH is an efficient heuristic for *on-line* GDS. Experiment results indicate that a MVH for *on-line* GDS is feasible and efficient, as it reduces the simulation execution time by 74% over *at-end* GDS. Further, the rules can be embedded in any ER model, without incurring major set up cost. The MVH is successfully built upon Reyes' one-variable heuristic.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER1[1]

INTRODUCTION

Simulation modeling enables the study of the stochastic behavior of systems, the testing of hypothesis that account for the observed behavior, and the use of these theories to predict future behavior (Centeno and Reyes, 1998).

One of the main reasons for the popularity of simulation systems comes from the assistance they give to decision makers in performing *what-if?* analysis. It is now widely accepted that supporting decision more efficiently requires assisting in *how-to?* analysis as well as *what-if?* analysis. *How-to?* analysis starts with a goal and determines one or several conditions to reach the goal. *How-to?* analysis is very important because many a times a system is simulated with a given goal or objective in mind (Shannon and Prakash, 1990). In the context of simulation, *how-to?* analysis can be performed by *goal-driven simulation* (Page *et al.,* 1999).

The research on on-line GDS has only been done with one variable, and these efforts have only given partial solutions. In the real world no system is analyzed on one variable. This research represents a multi-response approach for *on-line* GDS where multiple variables are introduced in solving the optimization problem.

---

[1] This thesis was supported in part by the ARISE Center under NASA grant NAG10-0212.

Response surface methodology and multi attribute utility theory were used to develop a two variable heuristic for GDS. 540 decision rules to change input parameters (number of nurses, doctors and beds) were developed for 15 different combinations of two out of six measures of performance; time in queue (for nurses, doctors and beds) and utilization (of nurses, doctors and beds). Two sets of experiments were done to verify the heuristic, and to compare *at-end* GDS to *on-line* GDS. Results from these experiments indicated that it is possible to perform multi variable *on-line* GDS with two measure of performance and it does yield 74% reduced simulation execution time over two variable *at-end* GDS, without changing the number of iterations significantly.

The rest of Chapter 1 gives the background, goal, and specific objectives of this research. Chapter 2 provides a literature review on Goal Driven Simulation. Chapter 3 describes the methodology used in this research. Chapter 4 details the results of the various experiments. Finally, Chapter 5 summarizes the results of this research effort and gives suggestions for further work.

1.1 Problem Statement

Simulation is the imitation of the operation of a real-world process or system over time (Banks, 1999). Activities involved in a simulation study require the modeler to possess knowledge in various disciplines, such as modeling, probability and statistics, computer programming and process analysis (Alsugair and Chang, 1994). Unless the modeler has extensive experience with the type of systems being modeled, s/he finds her/himself testing a large set of alternatives before a desired performance goal is achieved (Molina et al, 1996). Goal Driven Simulation (GDS) has emerged with the aim of driving the simulation model inputs towards the output specified by the user (Reyes, 1998).

For GDS to be very useful it must be automated; otherwise, it provides no new assistance to the modeler. Attempts have been made to automate the simulation modeling process (SMP) using knowledge-based systems. Examples are ROSS (Klahr et al., 1980), V-GOSS (Adelsberger and Neumann, 1985), and GDSS (Alsugair and Chang, 1994). These attempts have sought the automation of the SMP with different objectives. For example ROSS sought to integrate rules about objects and relationships and express them as a single rule, whereas GDSS sought to reduce the dependence upon a human expert and to fully utilize the capabilities of computers.

Goal Driven Simulation (GDS) *per se* have been approached in several ways. Most of the GDS attempts have been done by running the simulation to its completion, checking if the goal was achieved, and if not, rerunning the simulation with a new set of inputs. Thus, all the time and effort spent in trying to achieve the goal would be wasted. This approach has been called "*at-end*" GDS (Reyes, 1998). Reyes (1998) developed a heuristic that reduced the execution time of the simulation run by checking the outputs at several intermediate points. This approach has been called *"on-line"* GDS. The results of her research showed that *on-line* GDS is 66% faster than *at-end* GDS. Hence, it makes sense to focus only on *on-line* GDS for any future research.

Reyes tested her heuristic on an emergency room facility operating as a non-terminating system. Centeno and Forrest (1999) tested the portability of Reyes' heuristic on a manufacturing system. Jones (1999) tested the heuristic on a manufacturing system operating as a terminating system. Penaloza (1999) tried a different approach by developing a forecasting-based heuristic for *on-line* goal driven simulation. All of this research showed that on-line GDS was feasible and better than *at-end* GDS. However, these research efforts have only focused on one variable. Usually, no system is analyzed using one variable only. Many simulation studies and real world problems involve analysis of more than one variable. Therefore, it is important that *on-line* GDS be tested on multiple variables.

Even though most simulation packages are able to generate information regarding several measures of performance such as utilization of resources, time in system, time in queues, etc., GDS research has only used one of the measures of performance in the problem's objective function (Baesler and Sepulveda, 2000). This research represents a multi-response approach for on-line GDS where two variables are introduced.

A multi variable approach for on-line GDS (OLGDS) is a difficult task. In the literature, there are few attempts to solve multi variable simulation optimization problems. Clayton, Weber, and Taylor (1982) present a direct search approach for multi variable simulation optimization based on modified pattern search and goal programming with preemptive priorities. Rees, Clayton and Taylor (1985) proposed a procedure for obtaining satisfactory solutions to multiple variable simulation models using modified response surface methodology within a lexicographic goal-programming framework. Mollaghasemi (1994) presents an interactive approach for optimizing multi-variable simulation models based on the Geoffrion-Dyer-Feinberg (GDF) vector maximal algorithm. Yet, these efforts have only yielded partial solutions.

When introducing multi-variables to a system, the goal for each one of the variables may conflict with each other. For example, a goal for time in the queue (W_q) and a goal for number in queue (L_q) may never be contradictory, but a goal for utilization of resources (ρ) and a goal for time in queue may be contradictory. Specifically, if we seek to minimize W_q and to reduce the number in queue, one may either reduce the processing time ($1/\mu$) or add resources (c). This would also lead to a decrease of number in queue; hence, goals for W_q and L_q are not contradictory. But if we try to maximize the utilization of resources (ρ) and decrease the time in the queue, one may either increase demand (D) or reduce the number of resources (c). Either action may lead to an increase of ρ, and an increase in W_q and L_q because

If $W_q \rightarrow \mu$ Then $\rho \rightarrow 0$ and $L_q \rightarrow 0$.

If $D \rightarrow \infty$ Then $\rho \rightarrow 1$ and $W_q \rightarrow \infty$, and $L_q \rightarrow \infty$

Thus, if the user establishes a goal of a range (a, b) for W_q and a goal of (c, d) for the ρ, it is possible that the range (a, b) can only be achieved if the range (c, d) is violated in its lower bound (Reyes, 1998). As seen from Figure 1, if we have a goal of a range (2-5) hours for time in queue, and a goal of a range (0.9-0.95) for utilization, both goals cannot be met at the same time. Meeting one goal violates the other.

Exploring the inclusion of 2 or more variables is the driving force behind this effort. Some of the fundamental questions to answer are:

1. Can the goal for all variables be met? In other words, ascertain the feasibility of goal set (Γ).
2. Which variable should be favored in case Γ is infeasible? In other words, we need to test to see what variable to pull out to make Γ feasible. This process has to consider mathematical reasons and user preferences.
3. What input parameters should be changed to direct simulation results towards the goal set Γ?
4. By how much should we change the selected input parameters?

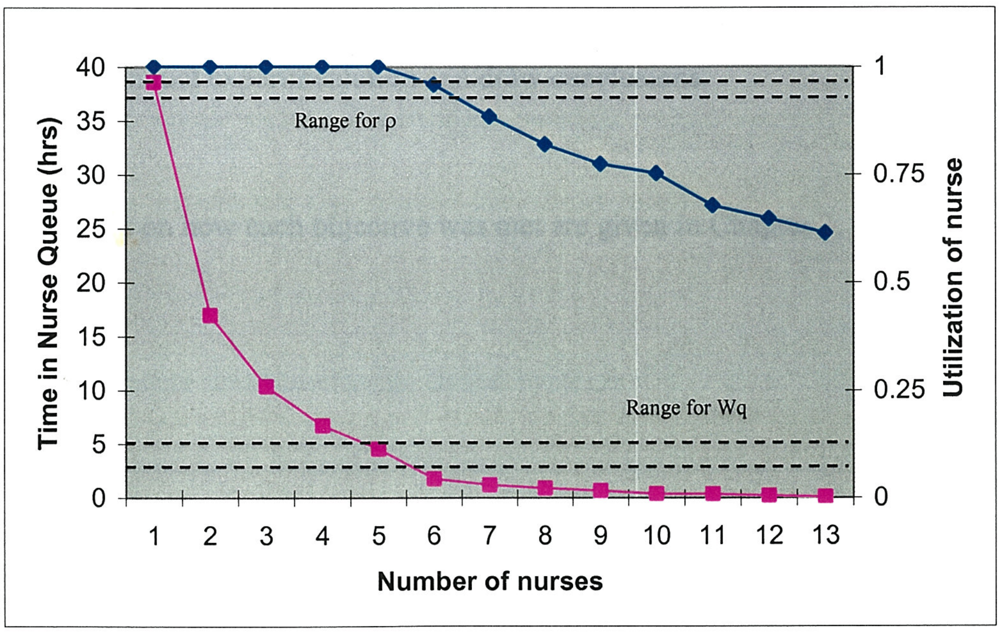


Figure 1: Example of Contradictory Goals

1.2 Goals and Specific Objectives

The goal of this investigation was to develop a methodology to extend the on-line GDS heuristic developed by Reyes (1998) to include two or more variables for non-terminating systems.

The specific objectives of this research were as follows:

1. Enhance the model of the non-terminating system used by Reyes, using the ARENA simulation software.
2. Study how the input parameters affect the outputs from the simulation model.
3. Develop a heuristic to handle two or more variables using VBA for ARENA.
4. Develop a set of experiments to test the heuristic.
5. Conduct experimentation and draw conclusions.

Details on how each objective was met are given in Chapter 3, 4 and 5.

CHAPTER 2

LITERATURE REVIEW

In this chapter, the efforts that have contributed to Goal Driven Simulation are reviewed using a historical perspective. The literature presents two main approaches to GDS:

1) Traditional, advanced computational algorithms and,

2) Knowledge-based heuristics.

In addition, response surface has been extensively used to determine the rules to change the inputs. Hence, this review is divided into three sections, one for each of these topics.

In this effort, the use of Multi Attribute Utility Theory (MAUT) was explored; thus, this chapter also reviews the fundamentals of MAUT.

2.1 Simulation and Knowledge Based Systems

Simulation is an indispensable problem-solving methodology for the analysis of many real-world problems (Jain, 1999). A simulation study is a search that relates the input parameters to response variables; however, this relationship is not thoroughly understood for all types of systems (Reyes, 1998). For that, one should be able to utilize the full range of inference, reasoning, and search methods that are available in the area of artificial intelligence to explain why a given sequence of events occurred and answer definitive questions such as "*can this event ever happen*?" and goal directed questions such as "*which events might lead to this event*?" Achieving this requires the use of search and other computational techniques such as Knowledge Based Simulation (Rothenberg, 1990). A knowledge-based system may be used for several functions, including the ones to automate the modification of input parameters in OLGDS.

In the process of analyzing simulation results, the decision-making requires significant human expertise and computer resources. To use simulation efficiently in the decision making process, the integration of knowledge-based systems with simulation is a must (Ford and Schroer (1987), Mellichamp et al., (1990), O'Keefe, (1986), Shannon, (1988)). By using knowledge-based system techniques, the skills required conducting simulation studies correctly and accurately, and knowledge for the simulation analysis process can be captured and stored (Chen et al., 2001).

One of the earlier attempts at integrating simulation with knowledge-based systems is ROSS, a Rule-Oriented Simulation System developed by the RAND corporation (Klahr, Faught, and Martins, 1980). The aim of this program was to specify facts and rules about objects and their relationships; query the system regarding these objects and relationships, and combine facts to express them as a single rule.

Another early effort was KBS (Fox and Reddy, 1982), which is a knowledge-based simulation system that incorporates object oriented programming (OOP) to describe the real world. It allows goals describing the performance criteria of model components to be attached to objects, and it informs the user whether the goals were met (Centeno and Standridge, 1992).

In the early part of the 1990's there was a resurgence in the use of KBS techniques, but with a much narrower scope and a more specific domain. For instance, SIMEX (Simulation Expert) is a prototype system developed to study the integration of simulation with expert knowledge in the construction industry domain (Touran, 1990). Although Reyes (1998) did not directly use a KBS, she developed a heuristic that combined three of O'Keefe's taxonomies: intelligent back-end, embedded, and parallel model. She developed a module that runs in the background, but is embedded in the model. The technique used by Reyes (1998) to predict the measures of performance was the confidence interval technique. She determined that the best stoppage interval was 10% of the simulation length. To halt the simulation execution at the desired stoppage points, VBA was incorporated to the simulation logic. Every time the simulation was halted, a 90% confidence interval was built around the mean of the measure of interest, which was time in the system. The heuristic in the VBA programming performed the comparison of the range of this interval with the range of the goal specified by the user. The heuristic assessment was based on the amount of overlap between these ranges.

2.2 Goal Driven Simulation

In GDS, the user states the goals to be met and the model drives itself to achieve them (Prakash and, Shannon 1989). GDS is an attempt to automate the traditional simulation modeling methodology (Penaloza, 1999). GDS capabilities include determining parameters to change, suggesting a rate of change, and testing these changes against a pre-established set of goals. In theory, it should incorporate the use of a knowledge-based expert system in conjunction with a simulation language to achieve a set of desired objectives. This knowledge enables the GDS tool to support the modeler even when the complexity of the system under study grows because it will guide him/her in how to modify parameters to achieve desired goals (Molina et al., 1996). They further state that a GDS system should have at least three components: a knowledge-based expert system, an open simulation language, and a programming language.

Initial approaches to GDS were based on automating the analysis of the results and changing the input parameters at the end of the run. The process was automatically repeated until the goal was reached. This approach is called *at-end* GDS. The main drawback of *at-end* GDS is that the total execution time tends to be long. Reyes (1998) investigated the feasibility of *on-line* GDS. The results of her research showed that *on-line* GDS is faster than *at-end* GDS. The main difference between *at-end* GDS and *on-line* GDS is the ability to halt the model execution when the statistics generated by the current version of the model do not achieve the goals set by the user.

Reyes tested her heuristic on an emergency room facility where conditions were assumed to be continuous. Centeno and Forrest (1999) tested the portability of Reyes' heuristic on a manufacturing system, which operated 24 hours a day, whereas Jones (1999) tested the heuristic on a manufacturing system operating as a terminating system. Penaloza (1999) tried a different approach by developing a forecasting based heuristic for on-line goal driven simulation. Results of her research showed that *on-line* GDS is a significantly better approach than *at-end* GDS when using a forecasting-based heuristic. Although there was not enough evidence to conclude that forecasting-based heuristics perform better than Reyes' (1998) confidence interval heuristics, the forecasting technique did address the data correlation issue.

Alsugair and Chang (1994) developed a system for civil engineering known as Goal Driven Simulation System (GDSS). GDSS is considered to be a problem solving approach in which the user specifies a model for a construction process and performance criteria for construction resources as well as a simulation goal to be achieved by conducting the simulation study.

2.3 Response Surface Methodology

Response surface methodology (RSM) is a collection of statistical and mathematical techniques useful for optimizing stochastic functions (Myers and Montgomery, 1995). This methodology is based on approximation of the stochastic objective function by a low order polynomial on a small sub region of the domain. The coefficients of the polynomial are estimated by ordinary least squares (OLS) applied to a number of observations of the stochastic objective function. To this end, the objective function is evaluated in an arrangement of points referred to as an experimental design (Kleijnen, 1998). Based on the fitted polynomial, the local best point is derived, which is used as a current estimator of the optimum and as the center point of a new region of interest (Fu, 1994).

Generally, the structure of the relationship between the response and the independent variables is unknown. A RSM algorithm comprises two phases. In the first phase, the response surface function is approximated by first order polynomials, until a polynomial is fitted that shows significant lack-of-fit, or until there is no direction of improved response anymore (Cochran and Cox, 1962). In the second phase, the objective function is approximated by a second order polynomial (Fu, 1994).

In RSM, the input parameters of the simulation model are called *factors*, whereas the stochastic output is called the *response*. The response can be represented graphically in a multi-dimensional space as contour plots that help visualize the shape of the response surface. It is assumed that a screening phase, in which factors that are considered unimportant are eliminated from the optimization problem, as well as possible transformations of the factors and the response, have already taken place. The objective is to optimize the variables of interest to produce the "best" output response.

In the literature there are few attempts to solve multi variable simulation optimization problems. The majority of these are focused on response surface methodology, utility theory, and interactive procedures where the decision-maker interacts with the model and leads the search.

Montgomery and Bettencourt (1977) applied a method based on the Geoffrion-Dyer interactive vector maximal algorithm. This approach makes use of the response surface methodology to estimate the objective functions equations, and then by solving a set of sub problems, the search direction is determined. Clayton, Weber and Taylor (1982) present a direct search approach for multi variable simulation optimization based on modified pattern search and goal programming with preemptive priorities.

Rees, Clayton and Taylor (1985) proposed a procedure for obtaining satisfactory solutions to multiple variable simulation models using modified response surface methodology within a lexicographic goal-programming framework. A lexicographic minimization is defined as a sequential minimization of each priority while maintaining the minimal values reached by all higher priority level minimizations (Tamiz et al., 1998). The most preferred goal is optimized first using the response surface approach. Then, an attempt to achieve the next highest ranked goal is made without violating the result obtained for the highest ranked goal. The same procedure is repeated for each one of the goals.

Matwiczak (1990) suggests an algorithm called AMOS (Automated Multiple Response Optimum-Seeker for Simulations). AMOS was designed to optimize a linear combination of simulation responses. This algorithm begins by asking the user to specify the problem parameters and uses a modified response surface methodology to arrive at the "best" solution.

Mollaghasemi, Evans, and Biles (1991) present an aggregation approach for multi variable simulation optimization. The method uses a multi-attribute value function representing the decision-maker preferences. Then a gradient search technique is used to find the optimum of the assessed function. Mollaghasemi and

Evans (1993) propose an interactive methodology based on a modification of Box's Complex Search (1965) for the stochastic optimization of multiple response simulation models. This approach involves conducting simulation experiments at each of the n+1 vertices of the complex (n is the number of decision variables), and computing the sample mean and sample standard deviation of each response. The decision maker is then presented with the results and is asked to identify the "worst" point. The "worst" point is then replaced by a point that is obtained by reflecting through the centroid of the remaining points by a distance β. This process contiues until convergence is achieved. Mollaghasemi and Evans (1994) proposed a modification of the multi-criteria mathematical programming technique called STEP method. This technique works in interaction with the decision maker who is asked, in each iteration, to identify the least satisfactory performance measure, which is then approved at the expense of other responses using a gradient search method. The process continues until the decision maker is satisfied with a solution or the number of available runs has been exhausted.

Teleb and Azadivar (1994) proposed an algorithm based on the constrained scalar simplex search method. The method works by calculating the objective function value in a set of vertices of a complex. The method moves towards the optimum by eliminating the worst solution and replacing it with a new and better solution. The process is repeated until a convergence criterion is met.

Mollaghasemi (1994) presents an interactive approach for optimizing multi variable simulation models based on the Geoffrion-Dyer-Feinberg (GDF) vector maximal algorithm. In this approach, the decision maker is asked to determine the trade-off ratios between a reference criterion and the remaining responses. This information in addition to the gradient estimate of each response is used to formulate a directional sub problem that after solving it will lead to the determination of the optimum direction. The process is repeated until the decision maker is satisfied with the solution.

Boyle (1996) presents a method called Pair wise Stochastic Cutting Plane (PCSCP). This method combines features from interactive multi-objective mathematical programming and response surface methodology. The method works by finding the center of the feasible region in the decision space and performing a design of experiments centered at that point. Interaction with the decision maker and cutting plane based techniques are used to determine the most preferred experimental point. Finally, formulating a new constraint based on the estimated gradient reduces the feasible region in the decision space. The process is repeated until the best compromise solution is found or terminating criteria are met.

Baesler and Sepulveda (2000) propose an approach that integrates a simulation model with a genetic algorithm heuristic and a goal-programming model. Their objective was to propose a general approach able find stochastically a global optimum for a multi variable simulation optimization problem.

Neddermeijer et. al, (2000) developed a framework for automated optimization of stochastic simulation models using response surface methodology. The framework is especially useful for automated optimization, in which all the settings of the algorithm have to be chosen at the outset of the optimization process.

2.4 The Use of Multi Attribute Utility Theory

Estimation of a user's interests in recommended solutions is very important. There are many approaches for estimating the user's interest. One approach is the Multi Attribute Utility Theory (MAUT) (von Neumann and Morgenstern, 1947). MAUT is an analytical method for making a decision concerning an action to take, given a set of multiple criteria upon which the decision is to be based. With (MAUT), a basic set of axioms is proposed to direct and restrict the way preferential judgments are made. Thus, MAUT enables structuring a complex problem in the form of a simple hierarchy and subjectively evaluating a large number of quantitative and qualitative factors in the presence of risk and uncertainty. The major strength of MAUT is its ability to deal with both deterministic and stochastic decision environments (Zionts, 1992).

According to Morrice et al. (1998), the use of utility theory ensures that any recommendation reflects:

- The interactions, if any, between measures of performance.
- The relative attractiveness of a specific level on a measure of performance.
- The relative attractiveness of performance on different measures.

A multi attribute utility function $u(X_k)$ has the form:

$$u(X_k) = \sum_{i=1}^{n} w_i u_i(x_i) \text{ for } k^{th} \text{ configuration}$$

where,

n = number of measures of performance

x_i = level of i^{th} measure performance

$u_i(.)$ = single attribute utility function over measure i

$0 \le u_i(.) \le 1$

w_i = weight for measure i

$\sum_{i=1}^{n} w_i = 1.$

The application of MAUT to complex problems usually involves the following steps (Edwards and Newman (1982), DeWispelare and Sage (1981)):

1. Identify the objectives of the decision and define the problem scope.
2. Define a finite set of relevant attributes affecting the decision outcome and structure them into a hierarchical tree.
3. Elicit preference information concerning the attributes from the decision maker and determine the relative importance of the attributes.
4. Develop the decision maker's utility function by establishing functional relationship between the attributes and the utility scores.
5. Compute the overall utility score for each decision alternative and rank alternatives in terms of overall utility scores.
6. Perform sensitivity analyses.

CHAPTER 3

EXPERIMENTAL FRAMEWORK

This chapter describes the conceptual framework for *on-line* GDS and the system used for experimentation. The work presented in this chapter satisfies proposed objective 1.

3.1 Nomenclature Used

W_{qi} = Average time in queue i

ρ_i = Average utilizaion of resource i

μ_i = Average rate of service at station i

λ = Average rate of arrival to the system

L_s = Average number in the system

L_{qi} = Average number in queue i

$$\mathrm{OF_i} = \text{Overlap flag} = \begin{cases} 1, \text{ if } \mathrm{G_i} \geq \Phi \\ 0, \text{ otherwise} \end{cases}$$

$$\text{Change} = \text{Change flag} = \begin{cases} 1, \text{ must change} \\ 0, \text{ no change, end of simulation} \end{cases}$$

$$\text{OneVariable} = \text{interest in OVH flag} = \begin{cases} 1, \text{ user is interested in OVH} \\ 0, \text{ otherwise} \end{cases}$$

m = Number of measures of performance

M_i = i^{th} measure of performance

n = Number of stoppage points : $n = (SL - T_o) / \Delta t$

P = Set of input parameters = $\{x_1, x_2, \ldots, x_l\}$

l = Number of input parameters

x_i = i^{th} input parameter

IPR_i = Range of allowable change for i^{th} input parameter

UGR_i = User's Goal Range for i^{th} measure of performance.

MCI_i = Model's 90% confidence interval for the i^{th} measure of performance.

Γ = User's goal set

Φ = Set of Threshold Proportion of MCI that falls in Rejection Region to *decide* if P should be changed.

G_i = Proportion of MCI that *actually* falls in Rejection Region for the i^{th} measure of performance.

Π = Proportion of simulation length (SL) at which to assess if the current P will hit the target.

SL = Simulation Length.

T_o = Warm up time.

Δt = Stoppage interval length.

K(Π) = Actual set of stoppage or check points.

UB(IPR) = Upper bound of IPR

newP = New Set of Input Parameters

3.2 Conceptual Framework of On-Line Goal Driven Simulation

The steps of a simulation study impacted by the use of GDS are *experimental design* and *analysis of outputs*. In the experimental design stage, the user inputs the goal and the system provides a solution that satisfies the goal, i.e. the user is given the values of the input parameters that meet the desired goal(s) after the model has automatically searched and changed them (Molina et al., 1996).

OLGDS research efforts seek to answer five principal questions (Reyes, 1998):

1. How should it be determined if the goal set is feasible?
2. When should the simulation run be halted to check the direction of the results?
3. How should the direction of the results be assessed?
4. How should the inputs be changed if the results are going in the wrong direction? and
5. Where in the timeline should the simulation be restarted if the inputs were changed?

These questions must be answered to fully support the framework shown in Figure 2 (Reyes, 1999). Previous efforts have provided answers to questions 2 and 3, and they have suggested how the others could be answered. Of the various heuristics, Reyes' offers a simple and robust one-variable heuristic. The summary of Reyes' heuristic is given in Table 1. A modified version of the nomenclature used in her heuristic is given in Section 3.1.

Reyes' effort of developing a heuristic for *on-line* GDS has provided a starting point towards achieving *on-line* GDS. Multi variables must now be introduced to the heuristic. Hence, this work uses:

- The confidence interval technique to estimate the direction of outputs.
- 10% of the simulation length as the Δt for checking the direction of outputs.
- VBA to build a 90% confidence interval around the mean of the measures of interest. This is done at every checkpoint.
- Two classes of measures of performance: time in the queue and utilization of resources. This yields a total of six specific measures of performance: time in nurse queue, time in doctor queue, time in bed queue, utilization of nurses, utilization of doctors, and utilization of beds.

These conditions yield:

P = {number of RN, number of MD, number of beds, number of registration clerk, number of triage nurse}

SL = 5300 hours	$T_0 = 1000$ hours	$\Delta t = 430$ hours
$\Pi = 10\%$	$n = 10$	$\Phi = 75\%$
$l = 3$	$m = 6$	

K(10%) = {1430, 1460, …,5300}

The heuristic programmed in VBA performs comparisons of the confidence interval ranges of these measures with the range of the goals specified by the user.

More details on the heuristic are given in Chapter 4.

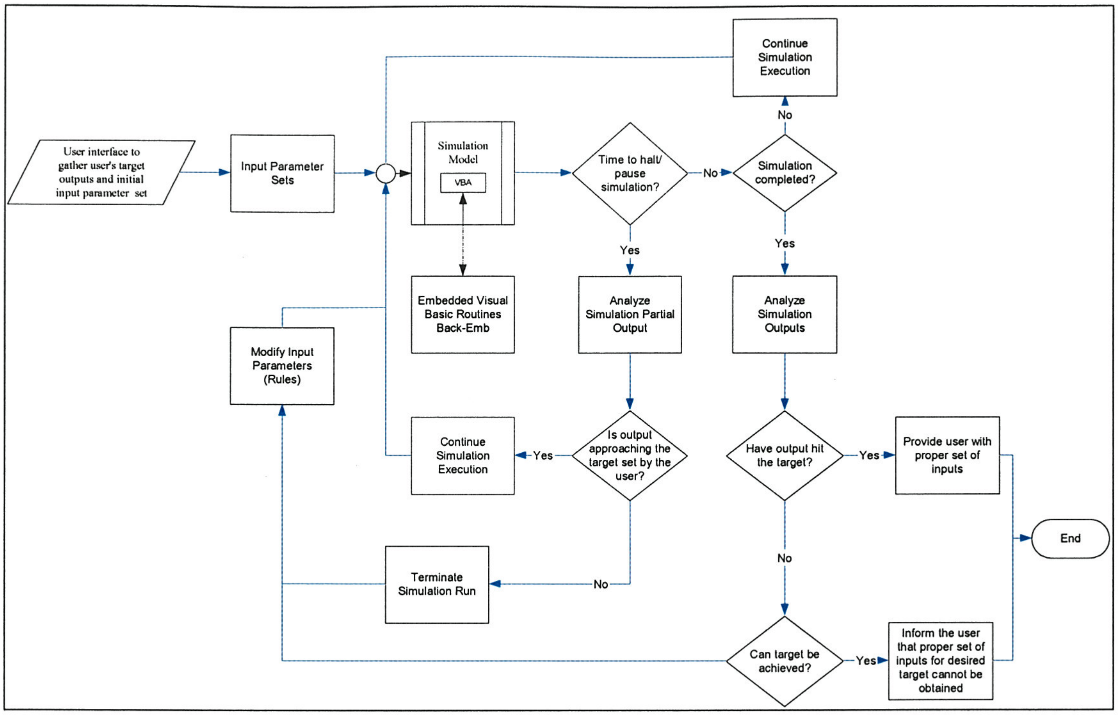


Figure 2: Framework for online Goal Driven Simulation (Reyes, 1999)

Table 1: Summary of One-Variable Heuristic (Reyes, 1998)

Let $\Phi = 75\%$ and $\Pi = 10\%$ $\Delta t = \Pi * SL$

$$K_j(\Pi) = \{(T_o + \Delta t), (T_o + 2\Delta t), (T_o + 3\Delta t), \ldots, (T_o + 10\Delta t)\}$$

1. At a stoppage point $K_j(\Pi)$, compute MCI on the mean of the measure of performance.
2. Compute G
3. If $G \geq \Phi$, then
   3.1. Stop the simulation execution
   3.2. Determine if input parameters can be changed
      3.2.1. If yes, change parameters and re-start simulation.
      3.2.2. If no, report to the user that goal cannot be met.
4. If $G \leq \Phi$, then resume the execution

If TNOW < SL and (SL - TNOW) > Δt, then go back to step 1.

Else, allow simulation to run through completion and and report results to user.

3.3 The ER Model

This work used the same emergency room (ER) facility used by Reyes. There are four different categories of patients (Table 2). The process followed by the patients is shown in Figure 3. Patients arrive to the ER by ambulance, by fire rescue, or by their own means. If a patient arrives by ambulance or by fire rescue, a bed has to be available upon his/her arrival; otherwise, the patient is taken to another hospital.

Patients arriving by their own means are evaluated by Triage Nurse and are assigned priority. Category 1 patients arriving by their own means are immediately placed on a bed after the Triage Nurse diagnosis them, and a registered nurse performs an initial evaluation. Category 2, 3 and 4 patients are escorted to a registration representative after the Triage Nurse assigns them the priority. Patients category 2 wait for an extra or regular bed to be available. Patients category 3 wait for an extra, a regular or a fast track bed. Patients category 4 wait for a fast track or a regular bed based on the availability of beds.

After registration, category 2, 3 and 4 patients wait in the waiting area until a bed is available for them. Category 2 patients can either have the extra bed or a regular bed. Category 3 patients can have fast track bed, regular bed or extra bed. Category 4 patients can either have a fast track bed or regular bed. After a bed is assigned to the patient he/she waits there until a registered nurse is available for initial evaluation. After the registered nurse has documented the initial evaluation, she/he informs the physician that a patient is waiting and of the results of the initial evaluation. The physician examines the patient and decides if the patient needs further testing and procedures. Also, the physician determines if the patient is to be admitted, observed in the ER, or discharged. If the patient is admitted he/she wait for a certain amount of time to get the paperwork done and the nurse or assistant from the other unit to come. If the patient is discharged he/she waits for a certain amount of time to get the paper work done. If the patient is observed he/she is checked by the Registered Nurse at a regular basis.

The inputs used for this model are summarized in Table 4. Details on how these inputs were established can be found in Reyes (1998). Several modifications were made to this model:

- The demand was increased from 4 patients/hour to 10 patients/hour to obtain a wider range of possible change of the input parameters. This change gives more options to derive rules for the heuristic.
- The fast track area was removed from the system, but the fast track bed was kept to make sure it resembles the real ER system.
- A final stage was added to the model for the decisions that are made regarding the patients condition. The patient is admitted, observed in ER or discharged.
- The NICKNAMES Element in ARENA 5.0 was used to make the model more understandable to humans and the modules from the COMMON panel were removed to ensure the model does not contain hidden or extra statements.

Table 2: Patients Category at Mercy's Hospital (Reyes, 1998)

Category	Condition
1. Emergent	Respiratory, trauma, cardiac, psyche, hemorrhage, etc.
2. Urgent	Emotionally disrupted behavior, abdominal pain, acute pain, etc
3. Non-Urgent	Nerves, sprains, chronic headache, eye infection, abrasion, etc.
4. Stable	Wound checks

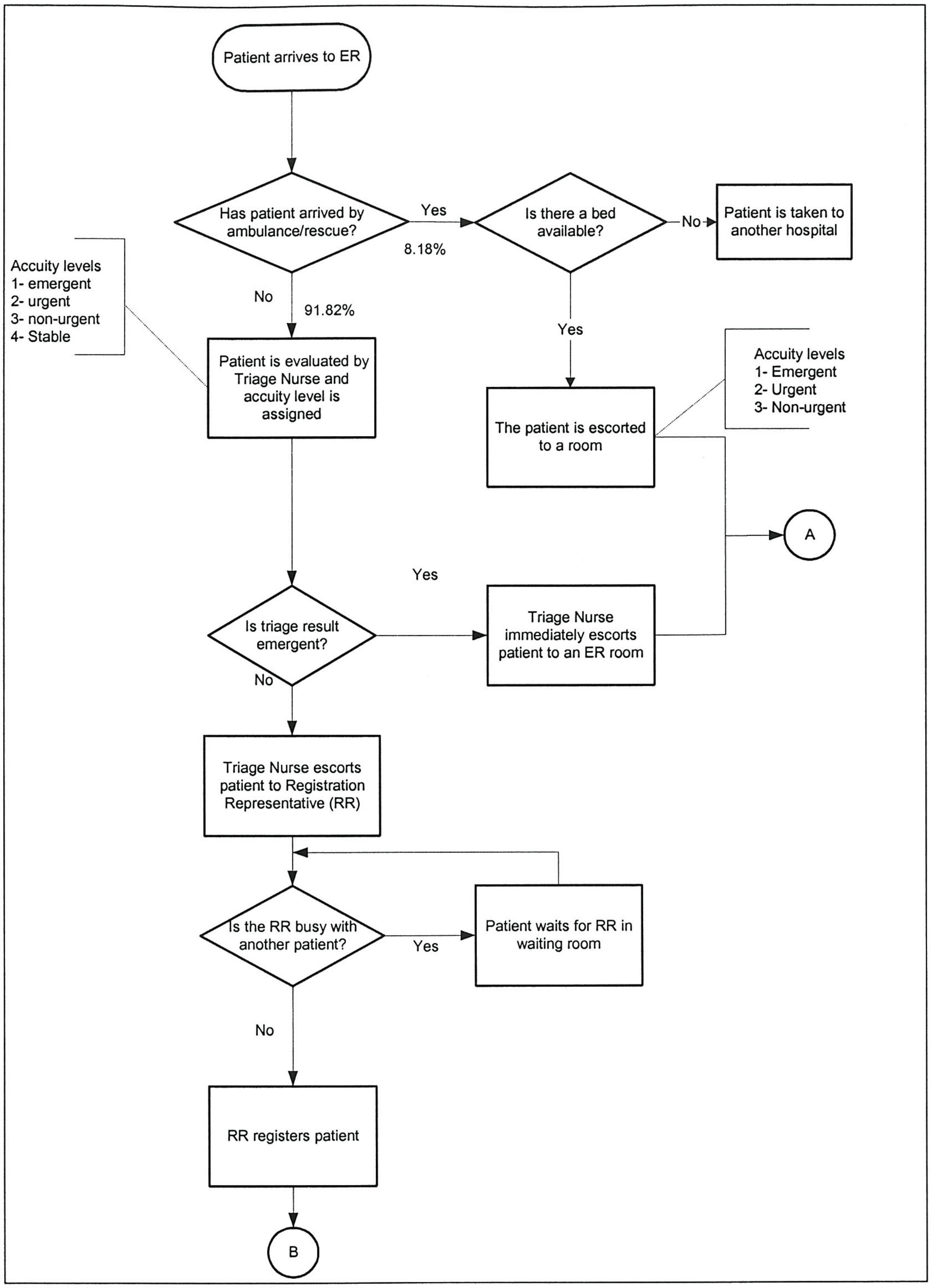


Figure 3: Patient Flow through the ER (Modified from Reyes, 1998)

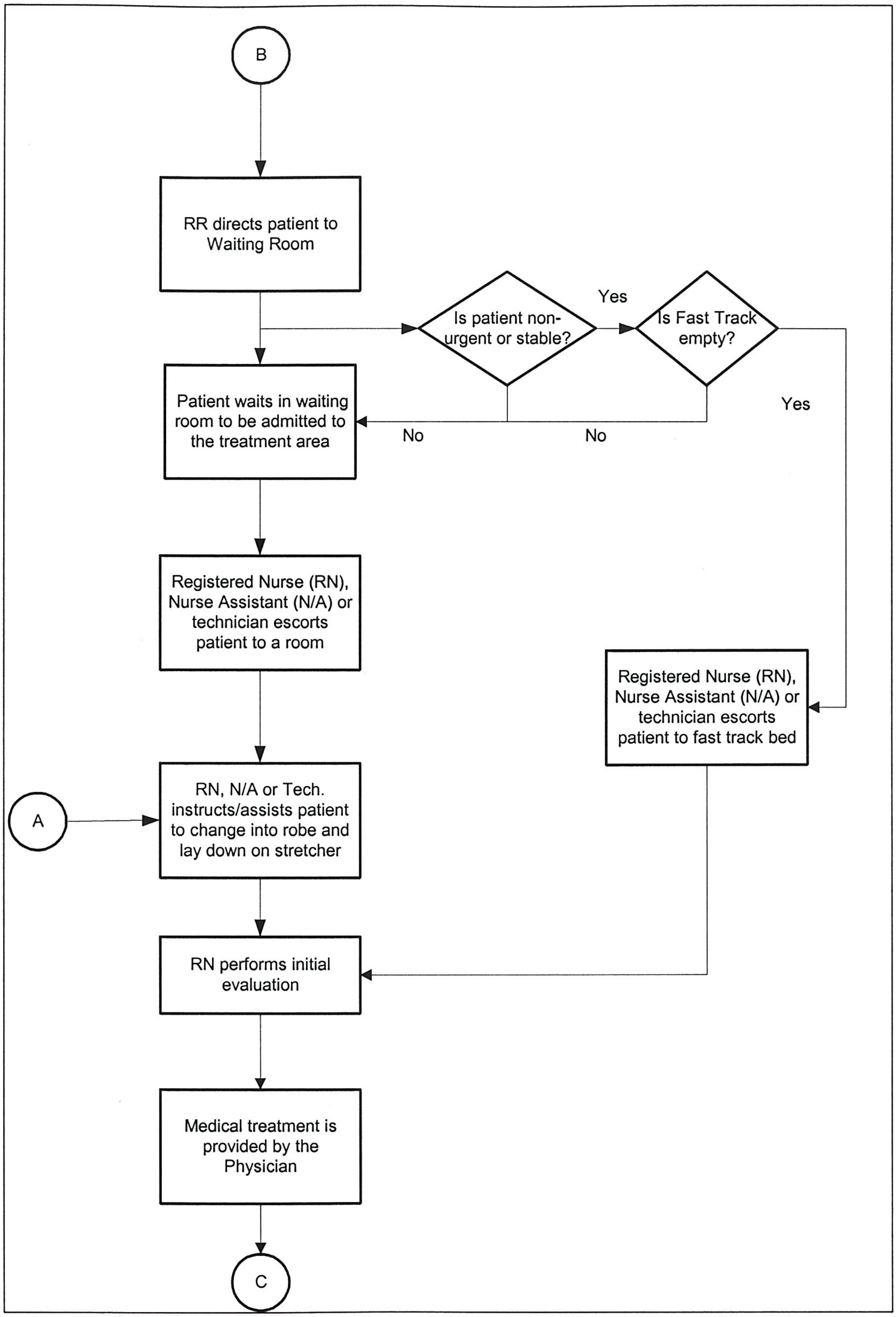


Figure 3b: Patient Flow through the ER (Continued)

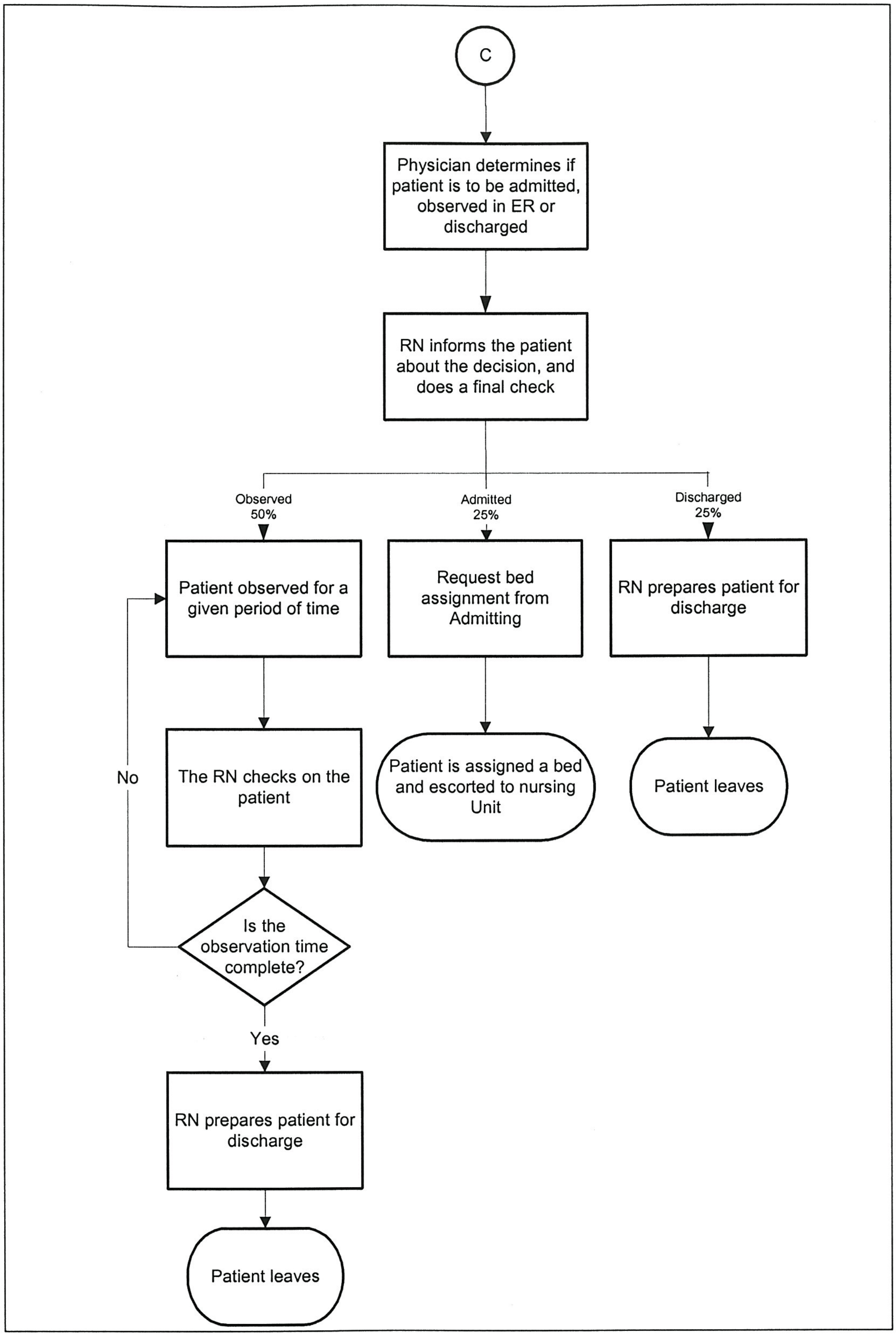


Figure 3c: Patient Flow through the ER (Continued)

Table 3: Initial Inputs to the Simulation Model

	Patient Category			
	1	2	3	4
Interarrival Time	**Expo (.10)**	**Expo (.10)**	**Expo (.10)**	**Expo (.10)**
Category Percentage (ambulance, car)	6% (3.56, 2.44)	44% (3.15, 40.85)	47% (1.47, 45.53)	3% (0, 3)
Triage Service Time	Uniform (.0833,.1667)	Uniform (.0833,.1667)	Uniform (.0833,.1667)	Uniform (.0833,.1667)
Registration Representative Service Time	Expo (.1167)	Expo (.1167)	Expo (.1167)	Expo (.1167)
Nurse Service Time	Uniform (.5,1)	Uniform (.5,.67)	Uniform (.33,67)	Uniform (.05,.5)
Doctor Service Time	Uniform (.17,.33)	Uniform (.167,.25)	Uniform (.0167,.083)	Uniform (.0167,.083)

3.4. Variables to Consider

Once the model was verified and validated, several experiments were done to find an appropriate goal set Γ. At this point, it was necessary to decide what specific measures to use in the study. Two classes of measures of performance were identified:

- Average Time in resource queue. (W_{qi})
- Average Utilization of resources. (ρ_i)

These measures were further broken down into:

W_{qMD} = Time in doctor queue ρ_{MD} = Utilization of doctors

W_{qRN} = Time in nurse queue ρ_{RN} = Utilization of nurses

W_{qbeds} = Time in bed queue ρ_{beds} = Utilization of beds

Queues within an ER are prioritized based on the patient sickness level. For example, Patients Category 1 have a higher priority over Patients Category 3. As a consequence, patients with low priority many a times have to wait long. An ER is perceived as a good one by its customer if the patient is given courteous and professional service relatively fast (Garcia et. al, 1995). Thus, it was a common sense to focus the attention on queue times. In addition, according to the effort done by Correa (1999), nurses and doctors are the most significant resources within an ER, and initial experimentation in this effort found that beds act as a correction factor. Consequently, it was decided to choose the above-mentioned six specific measures of performance.

CHAPTER 4

DEVELOPING A MULTIVARIABLE HEURISTIC

This chapter gives descriptions and findings of studying the relationships between inputs and outputs. The work in this chapter satisfies proposed objectives 2 and part of 3.

4.1 Procedure to Derive Rules

This research combines Reyes' (1998) one-variable heuristic with response surface methodology and multi-attribute utility theory concept to develop a multiple variable heuristic for GDS. Figure 4 represents the methodology used to develop the heuristic. Basically, the input parameters were identified and their ranges as well as the measures of performances to be included in the heuristic were decided. The simulation model was run for the given sets of inputs and the outputs for the desired measures of performances were collected. The outputs were first graphed and then linear regression analysis was performed. The results of these analyses are discussed in the next section.

The levels of the input parameters are given in Table 4. The number of Triage Nurse and Registration Clerk were kept constant at 2. These allow more patients to enter the system, so that more observation could be generated regarding the waiting time and the utilization of resources. There were 104 experiments run: 44

experiments were done with 4 levels for doctors (1-4), 11 levels for nurses (3-13), 1 level for beds (30), 1 level for registration clerk (2), and 1 level for Triage Nurse (2). 60 experiments were with 4 levels for doctors (1-4), 5 levels for nurses (6-10), 3 levels for beds (10-20-40), and one level for registration clerk and triage nurse (2). The results of the experiments are given in Appendix B. Length of the simulation run was 5300 hours. There is a warm up period of 1000 hours so that the system can get stabilized. The system and statistics are reinitialized after the warm up period. This is a non-terminating system, so there is only one replication. The outputs collected were the average value for the measure of performances from all the patients that came after the warm up period. Batching was not done because it was found that the outputs obtained without batching were equivalent to the outputs obtained with batching.

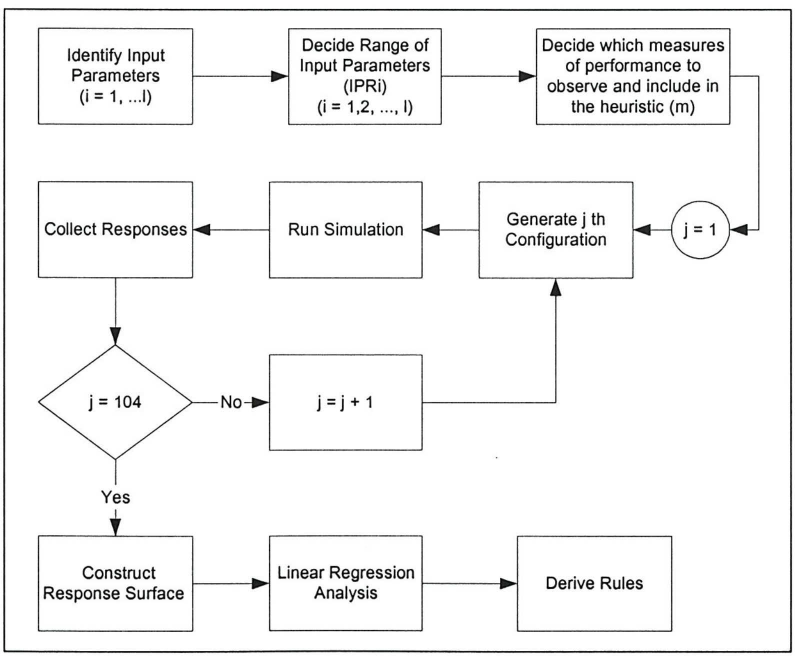


Figure 4: Rule Derivation Methodology

Table 4: Ranges of Input Parameters

i	Input Parameters	Levels	Range (IPR_i)
1	Number of doctors	4	1-4
2	Number of Nurses	11	6-10
3	Number of Beds	4	10, 20, 30 and 40
4	Number of Registration Clerk	1	2
5	Number of Triage Nurse	1	2

The Response Surface was graphed using MathCad Surface plot. After examining the surface graphs, it was found that the surface of the relationship between the independent variables and the dependent variables has patterns that could be modeled through an equation. Linear Regression Analysis was done to observe the relationship of the dependent variables: utilization of doctors, nurses, and beds and time in doctors' queue, nurses' queue and beds' queue with the independent variables: nurse, doctor, and bed. SPSS for Windows statistical software package was used to do the linear regression analysis. The significance level chosen was $\alpha = 0.05$ with $\nu 1 = 3$ and $\nu 2 = 100$. All the measure of performances were fit to a first order multiple linear regression model such as:

$$Y = \beta_o + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon$$

Where:

Y = measure of performance

x_1 = number of doctors

x_2 = number of nurses

x_3 = number of beds

ε = experimental error

β_i = ith regression coefficient

Notice that P_4 and P_5 are not part of the equations because preliminary analysis indicated that they act as stabilizing or correction factors, so they were set to a constant value.

A test for the significance of the regression was done to determine if there is indeed a linear relationship between the response variable Y and the regressor (independent) variables x_1, x_2 and x_3. The appropriate hypotheses are:

H_0: $\beta_1 = \beta_2 = \beta_3 = 0$

H_1: $\beta_j \neq 0$ *for at least one j and j = 1, 2, 3*

Which means that if we fail to reject the null hypothesis, there is no linear relationship between the response variable and the independent variables. However, if we reject it, there is a linear relationship between the response variable and at least one of the independent variables. In other words, rejection of H_0 implies that at least one of the independent variables (x_1, x_2, x_3) contributes significantly to the linear model. H_0 is rejected if $F_0 > F_{\alpha,k,n-k-1}$.

After the regression analysis was done, single attribute utility function of each of the measure of performances was found. Figure 5 shows the single attribute utility function of time in nurse queue. The MAU function was constructed from a weighted sum of the two single attribute utility functions. The weights assessed were 0.5 for each of the measures of performance selected. After obtaining the single attribute utility function of the measure of performances, the expected utility for of each one of the experimental configurations was calculated. Here the goal was to reach the highest expected utility for the users' goals.

The goal of this research was to reach both goals simultaneously. With multi attribute utility theory, it was realized that it is possible for the program to come up with an expected utility value that is the same as the goals, which has already met one of the goals but has not reached the goal for the second. Another problem is that in some cases there may be multiple solutions for the MAU function. These problems were overcome by asking the preference of the user, and the preferred *one* goal was given the weight = 1.

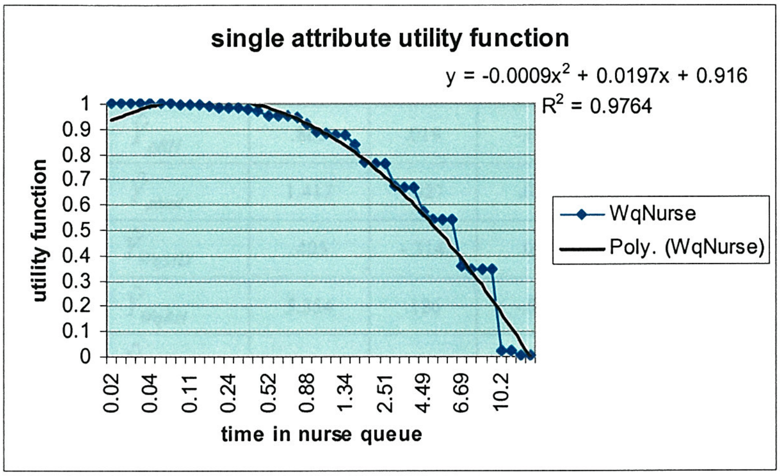


Figure 5: Single attribute utility function of W_{qRN}

4.2 Results from the Procedure to Derive Rules

The 104 experiments helped understand the behavior of the six selected measures of performance. Regression analyses yielded the first-degree polynomials shown in Table 5. The 95% confidence intervals for the regression coefficients, the F_0 and the R^2 values are given in Table 6.

The x_i and response variables are defined as follows:

x_1 = number of doctors x_2 = number of nurses x_3 = number of beds

$\rho_{MD} = \hat{Y}_{\rho md}$ = Doctor Utilization $W_{qMD} = \hat{Y}_{WqMD}$ = Time in Doctor Queue

$\rho_{RN} = \hat{Y}_{\rho RN}$ = Nurse utilization $W_{qRN} = \hat{Y}_{WqRN}$ = Time in Nurse Queue

$\rho_{bed} = \hat{Y}_{\rho bed}$ = Bed Utilization $W_{qbed} = \hat{Y}_{Wqbed}$ = Time in Bed Queue

Table 5: First degree Fitted Polynomials

Dependent Variable	X_0	X_1 (doctor)	X_2 (nurse)	X_3 (bed)
$\hat{Y}_{\rho md}$	.528	-.180	.026	.005
$\hat{Y}_{\rho RN}$	.885	.019	-.042	.009
$\hat{Y}_{\rho bed}$	1.412	-.027	.035	-.006
$\hat{Y}_{WqMD}$	.405	-.314	.028	.016
$\hat{Y}_{WqRN}$	5.354	.126	-.812	.095
$\hat{Y}_{Wqbed}$	6061.630	-96.88	-359.058	-67.036

As seen from Table 6, for each of these tests $\mathbf{F}_0 > \mathbf{F}_{.05;3;100} = 2.70$. This means that the first-degree polynomials explain the relationship between variables. Furthermore, the R^2 values are high, implying that the linear regression is explaining the variation in the dependent variable. The only low R^2 is that of $\hat{Y}_{qmd}$, for which $R^2 = .429$. Even though this value is not high, it is safe to say that the regression model explains some of the variation. It is not expected that this equation fit the data perfectly, but hopefully it is a good approximation. Hence, it can be said that all models fit the data reasonably well.

These results confirm previous findings, and as expected, there is a strong linear relationship among:

✓ Number of nurses, doctors and utilization of doctors

✓ Number of nurses, doctors and utilization of nurses

✓ Number of nurses, doctors and time in nurse queue

And there is a moderate linear relationship among:

✓ Number of nurses, doctors and utilization of beds

✓ Number of nurses, doctors and time in doctor queue

✓ Number of nurses, doctors and time in bed queue

Table 6: 95% confidence intervals for the regression coefficients, F_0 and R^2

Dependent Variable	β_0		β_1		β_2		β_3		F_0	R^2
	UL	LL	UL	UL	LL	LL	UL	LL		
$\hat{Y}_{utilmd}$	.440	.616	-.195	-.164	.019	.034	.003	.007	195.3	.854
$\hat{Y}_{utilRN}$	.809	.962	.006	.033	-.048	-.035	.008	.011	103.7	.757
$\hat{Y}_{utilbed}$	1.331	1.493	-.042	-.013	-.041	-.027	-.007	-.004	53.03	.614
$\hat{Y}_{qmd}$	*-.042*	*.851*	*-.394*	*-.234*	*-.011*	*.067*	*.007*	*.025*	*25.02*	*.429*
$\hat{Y}_{qRN}$	4.042	6.665	-.110	.362	-.926	-.699	.069	.122	84.42	.717
$\hat{Y}_{qbed}$	5320	6802	-230.8	36.51	-423.3	-294.7	-81.91	-52.17	766.6	.672

A further screening of the polynomials given in Table 5 confirms intuitive expectations. For example, theoretically, W_{qRN} and ρ_{RN} should decrease when adding units of resource nurse; the polynomials confirmed that W_{qRN} and ρ_{RN} indeed decrease when the number of nurses is increased. In fact, every unit of nurse decreases the waiting time by 14.56% (desirable) and the utilization by 4.7% (not desirable). It is curious to see that by adding one doctor, the waiting time for the nurse increases by 2.3%: *how can this be explained*? Since more doctors are available to treat patients, there are more patients in the treatment area requiring the services of the nurses. Hence, when the number of nurses does not increase, the waiting time for nurse will increase. The same behavior is observed when one bed is added: the waiting time for nurse increases by 1.8%. More beds mean more patients in the treatment area that need the services of nurses.

Another example would be the decrease in W_{qMD} and ρ_{MD} when increasing the number of doctors. Every unit of doctor decreases W_{qMD} by 70% (desirable) and ρ_{MD} by 32% (not desirable). Since more doctors are available to treat patients, there are fewer patients in the treatment area requiring the services of the doctors. Hence, when the number of nurses or beds does not increase, the waiting time for doctors will decrease, but the utilization of doctors also decreases. Again, increasing the counterpart resources, nurses and beds, increases the waiting time for doctors by 4.7% and 0.8% respectively.

Appendix C gives the graphical representations of the actual response surfaces. All the surfaces are filled with a color map. A color map is a file consisting of data, which match different levels of a plot to different colors. MathCAD converts all the data into an integer between 0 and 255. Any value that is not an integer is truncated. Any value that is not between 0 and 255 is increased to 0 or decreased to 255. There are usually 256 rows in a color map. Each row of values specifies a color consisting of red, green, and blue to which a grayscale value is mapped. Red color matches with the high values in the data, blue matches with the low values, green goes in the middle.

4.3 Discussion of the Results

The increase in the resources does not affect the measures of performance equally. As can be seen from Table 7, a unit increase in number of doctor, increases W_{qRN} by 2.3%, but we see a decrease of 77.53% in W_{qMD}. W_{qRN} is the performance that is most affected by an increase in resource nurse, while W_{qMD} is the most affected by an increase in resource doctor. The increase in doctor improves W_{qMD} 37 times more than it worsens W_{qRN}, while also improving W_{qbed} slightly. An increase in nurse improves W_{qRN} three times more than it affects W_{qMD}, while also improving W_{qbed} significantly. If we look at the polynomials from a different point of view, it is possible to rank the resources from 1 to 3 according to the absolute effect on each of the measure of performances (positive or negative). Table 8 shows that nurses are the most crucial resource for 4 of the 6 measures of performance, and bed is the least crucial for five of the 6 measures performance. Even though beds do not affect the measures of performance very much, they seem to act as an adjusting factor.

Table 7: Effect of single resource changes

Dependent variables	**(x_1, x_2, x_3) (0, 0, 0)**	**(x_1, x_2, x_3) (1, 0, 0) Doctor**	**(x_1, x_2, x_3) (0, 1, 0) Nurse**	**(x_1, x_2, x_3) (0, 0, 1) Bed**
$\hat{Y}_{\rho md}$	.528	-34%	4.92%	**9.46%**
$\hat{Y}_{\rho RN}$	.885	**2.15%**	-4.75%	1.02%
$\hat{Y}_{\rho bed}$	1.412	-1.91%	**2.48%**	-0.42%
$\hat{Y}_{WqMD}$	.405	**-77.53%**	6.91%	3.95%
$\hat{Y}_{WqRN}$	5.354	2.35%	**-15.17%**	1.77%
$\hat{Y}_{Wqbed}$	6061.630	-1.6%	**-5.92%**	-1.11%

Table 8: Ranking of the effects

Measure of Performance	Change		
	Doctor	**Nurse**	**Bed**
Utilization of Doctors	1	3	2
Utilization of Nurses	2	1	3
Utilization of beds	2	1	3
Time in Doctor Queue	1	2	3
Time in Nurse Queue	2	1	3
Time in Bed Queue	2	1	3

Table 9: Effect of multiple changes of resources

Dependent variables	(x_1, x_2, x_3) (0, 0, 0)	(x_1, x_2, x_3) (1, 1, 0) (MD and RN)	(x_1, x_2, x_3) (1, 0, 1) (MD and Bed)	(x_1, x_2, x_3) (0, 1, 1) (RN and Bed)	(x_1, x_2, x_3) (1, 1, 1) (MD, RN and bed)
$\hat{Y}_{\rho md}$	.528	-29.17%	-33.14%	**5.87%**	-28.22%
$\hat{Y}_{\rho RN}$	.885	1%	**3.16%**	-3.73%	-1.58%
$\hat{Y}_{\rho bed}$	1.412	0.6%	-2.34%	**2.05%**	0.14%
$\hat{Y}_{WqMD}$	.405	-70.62%	**-73.58%**	10.86%	-66.67%
$\hat{Y}_{WqRN}$	5.354	**-17.5%**	4.13%	-13.39%	-11.04%
$\hat{Y}_{Wqbed}$	6061.630	-7.5%	-2.7%	-7.03%	**-8.63%**

Now, if the polynomials are examined for simultaneous change (Table 9) we can see that adding a unit of doctor and a unit of nurse decreases ρ_{MD} by 29.17% (not desirable), but it also decreases W_{qMD} by 70.62% (desirable). The addition of a unit of nurse and a unit of bed decreases W_{qRN} by 13.39% and at the same time decreases ρ_{RN} by 3.73%.

The comparison of *single resource change* versus *multiple resource change* gives a clear idea of how multiple resource addition can be useful. For example, adding a unit of nurse decreases W_{qRN} by 15.17%, while adding a unit of nurse and a unit of doctor decreases W_{qRN} by 17.5%. Adding one unit of nurse increases ρ_{MD} by 4.92%, while adding a unit of nurse and a unit of bed increases ρ_{MD} by 5.87%. As seen from these examples, multiple resource changes may be used for a faster achievement of a desired goal.

From the one variable perspective, these results lead to the following conclusions:

- A unit of doctor is worth adding if W_{qMD} needs to be decreased. The cost is a medium decrease in ρ_{MD}.
- A unit of nurse is worth adding if W_{qRN} needs to be decreased. The cost is a slight decrease in ρ_{RN}.
- A unit of bed is rarely worth adding once a stabilizing number of beds have been set.

From the two variable simultaneous perspective, the following conclusions can be derived:

- A unit of doctor and a unit of nurse are worth adding if W_{qRN} or W_{qMD} need to be decreased. The cost is a high decrease in ρ_{MD}. The benefit is a slight increase in ρ_{RN} and ρ_{bed}, and a very high decrease in W_{qMD} and a significant decrease in W_{qRN}.
- A unit of doctor and a unit of bed are worth adding if W_{qMD} needs to be decreased. The cost is a high decrease in ρ_{MD}, but the benefits are a

slight increase in ρ_{RN}, a slight decrease in W_{qRN} and W_{qbeds}, and a very high decrease in W_{qMD}.

- A unit of bed and a unit of nurse are worth adding if ρ_{MD} needs to be increased or W_{qRN} needs to de decreased. The cost is a slight decrease in ρ_{RN} and a medium increase in W_{qMD}. The benefits are a slight increase in ρ_{bed}, a medium decrease in W_{qRN} and in W_{qbeds}.
- A unit of doctor, a unit of nurse and a unit of bed are worth adding if W_{qMD}, W_{qRN}, and W_{qbeds} need to be decreased. The cost is a high decrease in ρ_{MD} and a slight decrease in ρ_{RN}. The benefits are a slight increase in ρ_{beds}, a very high decrease in W_{qMD} and a medium decrease in W_{qRN} and W_{qbeds}.

Some of the 3-D surfaces looked non-linear despite the fact that they passed the regression fit test. To explain this, 2-D graphs were created for each independent variable. Analysis of these graphs confirms that some of the factors do have a strong relationship. Table 10 shows that number of doctors has a strong linear relationship with only ρ_{MD}, while number of nurses has a strong linear relationship with ρ_{RN}, ρ_{beds}, W_{qRN}, and W_{qbeds}. Number of beds does not have a strong linear relationship with any of the measures of performance. In depth analysis of the exploration of non-linear relationships is left for a future study.

Table 10: Strength of Linear Relationship

	ρ_{MD}	ρ_{RN}	ρ_{beds}	W_{qMD}	W_{qRN}	W_{qbeds}
# of Doctors	Strong	Weak	Weak	Moderate	Weak	Weak
# of Nurse	Weak	Strong	Moderate	Weak	Strong	Strong
# of Beds	Weak	Moderate	Weak	Weak	Weak	Moderate

4.4 Derivation of Rules to Change Input Parameters

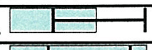
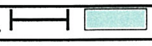
Rules were derived for the combination of two measures of performance out of six, resulting in 15 possible pairs (Table 12).

$$C_2^6 = \frac{6!}{2!4!} = 15$$

There are six different scenarios for the confidence interval of each measure of performance (Table 11). We are interested in two of the measures simultaneously, adding up to 36 scenarios for each combination given in Table 12. This means there is a total number of 540 possible rules for the 15 combinations.

The scenarios given in Table 11 must be interpreted according to the measure of performance. Scenario one, for instance, represents a better than desired situation for time in queue, but it is a worse than desired situation for utilization of a resource. This interpretation was used in deciding what to change. Furthermore, the information derived in Section 4.3 determined what input parameters to change for each one of these scenarios. However, it did not provide a solid mathematical basis in regard to *by how much* to change the parameters. Each scenario was evaluated using the information (rounded of) from Table 7 and Table 9.

Table 11: The Scenarios for each measure of performance

Scenario	**Condition**
1. 	$B < C$
2. 	$C < B$ and $B < D$
3. 	$A < C$ and $B > D$
4. 	$A > C$ and $D > B$
5. 	$A > C$ and $D < B$
6. 	$C < B$

Where:

A B : Model (MCI) C D : Goal (UGR)

Table 12: Combinations for two goals

Combination #	Goal-1	Goal-2
1	W_{qRN}	ρ_{RN}
2	W_{qRN}	ρ_{MD}
3	W_{qRN}	ρ_{bed}
4	W_{qRN}	W_{qMD}
5	W_{qRN}	W_{qbed}
6	W_{qMD}	ρ_{RN}
7	W_{qMD}	ρ_{MD}
8	W_{qMD}	ρ_{bed}
9	W_{qMD}	W_{qbed}
10	W_{qbed}	ρ_{RN}
11	W_{qbed}	ρ_{MD}
12	W_{qbed}	ρ_{bed}
13	ρ_{RN}	ρ_{MD}
14	ρ_{RN}	ρ_{bed}
15	ρ_{MD}	ρ_{bed}

For example consider the combination of W_{qRN} and ρ_{RN} when they are in Scenario (1,1) (Table 13). In this scenario, W_{qRN} is already better than the UGR; hence, the only concern is how to meet the goal for ρ_{RN}, i.e. how to increase ρ_{RN}. ρ_{RN} can be increased by:

1. adding a unit of doctor (≈ 2%)
2. adding a unit of bed (≈ 1%)
3. adding a unit of doctor and nurse (≈ 1%)
4. adding a unit of doctor and bed (≈ 3%)

Instinctively, the decision would be to add a unit of doctor and a unit of bed since it would increase ρ_{RN} faster, but the effect of this decision on W_{qRN} must be examined as it may bring W_{qRN} to an undesirable level.

Now, focusing only on two of the four options:

1. adding a unit of doctor ($\approx$ 2%)
2. adding a unit of doctor and bed ($\approx$ 3%)

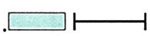
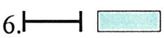
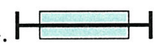
The judgment that must be made is whether option 1 with its 2% increase in ρ_{RN} and its 2% increase in W_{qRN} is better than option 2, with its 3% increase in ρ_{RN} and its 4% increase in W_{qRN}. Although W_{qRN} is better than UGR at the moment, the 4% increase of option 2 may bring W_{qRN} to being worse than UGR twice as fast as option 1 could with its 2% increase. Running this risk is not worth it given that there is only 1% benefit in ρ_{RN}. Hence, option 1 is selected as the decision rule for this scenario.

For another example consider the combination of W_{qRN} and W_{qMD} when they are in Scenario (6,4) (Table 13). The decision made here is to add a unit of doctor and 2 units of nurse. Here W_{qMD} has met the goals, so the effects that could decrease W_{qRN} are considered. W_{qRN} can be decreased by:

1. adding a unit of nurse ($\approx$ -15%)
2. adding a unit of nurse and a unit of doctor ($\approx$ -18%)
3. adding a unit of nurse and a unit of bed ($\approx$ -13%)
4. adding a unit of nurse, a unit of doctor and a unit of bed ($\approx$ -11%)

The decision here would be to add a unit of nurse and a unit of doctor because it decreases W_{qRN} faster. This change will decrease W_{qMD} by 71%. Since we have already met the goal for W_{qMD} having this much decrease is not necessary. So instead of adding a unit of doctor and only a unit of nurse, two nurses can be added for a faster decrease on W_{qRN}, while securing a better value for W_{qMD}.

Table 13: Example of Analysis for Deciding Rules

W_{qRN}	ρ_{RN}	**Possibilities**	W_{qRN}	ρ_{RN}
1. 	1. 	ADD MD	2%	2%
		ADD BED	2%	1%
		ADD MD + RN	-18%	1%
		ADD MD + BED	4%	3%
		CHOOSE: ADD 2 MD		
W_{qRN}	W_{qMD}	**Possibilities**	W_{qRN}	W_{qMD}
3. 	1. 	ADD RN	7%	5%
		ADD BED	4%	9.5%
		ADD RN + BED	11%	6%
		CHOOSE: OVH		
W_{qRN}	W_{qMD}	**Possibilities**	W_{qRN}	W_{qMD}
6. 	4. 	ADD RN	-15%	7%
		ADD MD + RN	-18%	-71%
		ADD RN + BED	-13%	11%
		ADD MD + RN + BED	-11%	-67%
		CHOOSE: ADD 1 MD + 2 RN		

In some scenarios, the MCI gives better results than the UGR, which would probably be desired by the user. In those cases, it was decided that the goal set (Γ) is met, and the user is informed that s/he got even better results than s/he asked for. An example for these scenarios would be for Wq_{RN}. If Wq_{RN} is in scenario number 1 (Table 11), it means that the waiting time for nurses is less than the one the user wanted. Another example is ρ_{RN} in scenario number 6, which means that utilization of nurses is greater than the one expected by the user. However, if the user wants to strictly meet the goal, the decision rules need to be modified. The latter has been left upen for future work.

The one variable heuristic is the only solution in some scenarios. If the MVH determines that both goals cannot be met simultaneously, the OVH (one variable heuristic) is activated. An example of such a case is for the combination of W_{qMD} and ρ_{MD}, in Scenario (3,1) (Table 13). In this case W_{qMD} has already met the UGR but ρ_{MD} needs to be increased. ρ_{MD} can be increased by:

1. adding a unit of nurse ($\approx$ 5%)
2. adding a unit of bed ($\approx$ 10%)
3. adding a unit of nurse and a unit of bed ($\approx$ 6%)

All of these possibilities also cause W_{qMD} to increase, which is not desirable. So the decision made is to activate the OVH. Obviously, this can only be done if the user is interested in meeting the goal for at least one variable. In the cases that the OVH is activated the decision to change the input parameters had to be made depending on the single measure of performance. The summary for each one of the 540 scenarios is given in Appendix E.

4.5 Development of the Heuristic

The heuristic combines the procedure developed by Reyes (Table 1) with the rules derived in Section 4.4. For the heuristic to yield its benefits, it has to be embedded in the simulation model, or it has to be integrated to the model in some fashion. Figure 6 offers a generic framework for implementing the heuristic and

Table 14 gives the heuristic. To distinguish between both heuristics, Reyes' is named One Variable Heuristic (OVH), whereas the one developed in this effort is called Multi Variable Heuristic (MVH).

The framework given here is general, and it can be applied to other simulation packages. One must start with an appropriate model of the system under study. Then the simulation model and a programming language must be used together to build a bridge between the model and the GDS heuristic. The framework requires an interface that acts as a link between the user and the simulation model through which the user is able to specify the allowable range for input parameters (IPR_i), the goal set (Γ), and the desired range (UGR_i) for the measures of performance (M_i).

Table 14: Steps of Multivariable Heuristic

STEP 1	Get from user the measures of performance (M_i). i = 1, 2, …, m
STEP 2	Get from user goal ranges: For i = 1 to m: Ask user for $LBUGR_i$ (M_i) and $UBUGR_i$ (M_i) Next i
STEP 3	Ask user if s/he is interested in OVH: If Yes Then Set OneVariable = 1 Else Set OneVariable = 0
STEP 4	Set experimental constants: Set n = 10 (number of stoppage points); Set Π = 10%; Set ϕ = 0.75 Set K (Π) = {1430, …,5300 }
STEP 5	Load decision rules for MVH and OVH

Table 14 b: Steps of Multivariable Heuristic (Cont.)

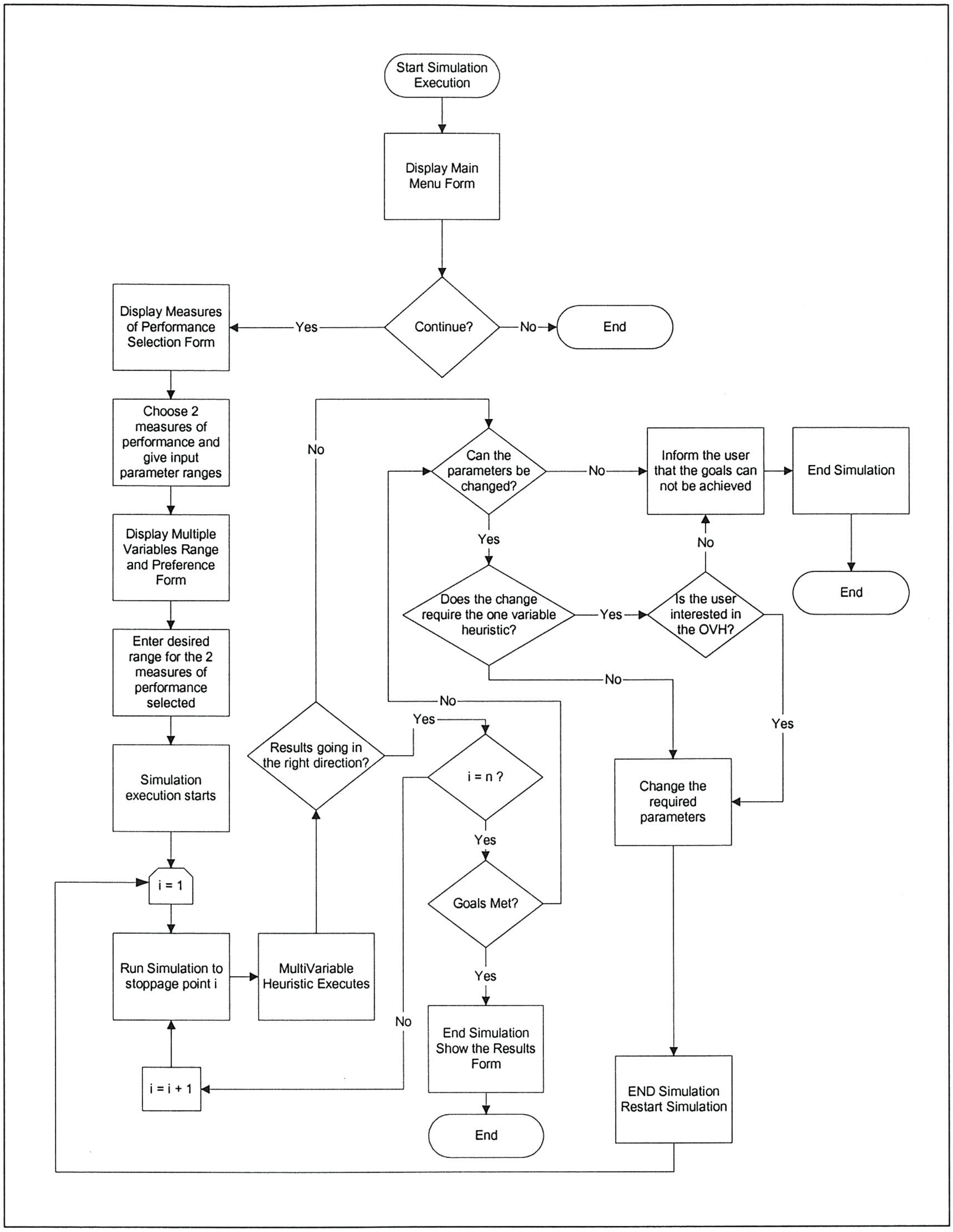
STEP 6	Determine if change is needed:
	j = 1; change = 0
	While $j \leq n$ and change = 0
	For i = 1 to m
	Compute 90% MCI_i on Avg of M_i and Compute G_i for M_i
	If $G_i \geq \phi$ Then Set $OF_i = 1$ Else Set $OF_i = 0$
	Next i
	If $OF_i = 1\ \forall_i = 1, 2, \ldots, m$ Then
	Change = 1; Change is needed
	Else
	If $OF_i = 0\ \forall_i = 1, 2, \ldots, m$ Then
	If j = n Then
	Change = 2; Change is not needed
	Else
	Continue until next stoppage point, j = j + 1
	End If
	Else
	If For 50 % or more of M_i have $OF_i = 1$ Then
	Change = 1; Change is needed
	Else
	Continue until next stoppage point, j = j + 1
	End If
	End If
	End If

STEP 7	If Change = 1 Then
	Stop Simulation
	Compute the number of the Decision Rule to use for MVH
	If P change requires OVH Then
	If OneVariable = 1 Then
	Compute the rule number of the Decision rule for OVH
	Else
	Continue until next stoppage point, j = j + 1
	End If
	End If

STEP 8	Change Parameters
	If NewP $\leq$ UB(IPR) Then
	P = NewP, Restart simulation, j = 1 Go To STEP 4
	Else
	Go To STEP 9
	End If
	End If
	Loop

STEP 9	Show the appropriate results or information.
STEP 10	End the simulation run

Figure 6: Multivariable On-Line GDS Framework



CHAPTER 5

IMPLEMENTATION AND TESTING OF THE HEURISTIC

This chapter describes the implementation and experiments done for testing the heuristic as well as the experiments to compare *at-end* GDS and *on-line* GDS with multivariable heuristic. The work in this chapter satisfies proposed objectives 3, 4, and 5.

5.1 Enhancing the Model's User Interface

Several changes were made regarding the user interface:

- In the **Main Menu** form, the "Continue with Current Goals" button was deleted to make the program more automated. Now the form has only there is only options "Enter New Simulation Goals" and "Cancel and Exit".
- The **Measure of Performance Selection** form was modified, so that the user can choose any two of the six measures of performance proposed. In the same form, the user is allowed to give the range for resources: nurse, doctor and beds.
- The **Multiple Variables** form was added to collect the range for the selected measures of performance. The user is also asked to specify his/her preference with respect to those goals.

- The **Results** form has also been modified, so that the user can see the results of the measures of the performance s/he specified as well as the number of resources that satisfies the goal, and the number of iterations it takes to reach the goals.

The revised model was developed using the commercial simulation environment ARENA 5.0. The text version of the model and experimental frames are given in Appendix A. GDS does not depend on ARENA, but ARENA offers VBA to enable *on-line* GDS. Hence, VBA was selected to program the heuristic and for the development of the user interface.

Figure 7 to Figure 10 show the user interface of this research. The first form that the user sees after s/he hits the "**Go**" button is the **Main Menu** form (Figure 7). After the **Main Menu** form, the user has to indicate the allowable range for each resource (nurse, doctor and beds), as well as selecting any two of the six measures of performance given in the **InputBounds** form (Figure 8). If the user selects more than two of the measures of performance a message box pops up telling the user to select only two measure of performance. In the case that the user selects less than two, the user is prompted to select two measures of performance two continue. The program also checks if the ranges entered for the resources are valid.

After the user clicks **Done** button, **Multiple Variables** form is shown (Figure 9). This is the last step before the simulation execution starts. Here, the user sees all the measure of performances disabled except for the ones s/he selected. The user is also asked to specify preferences among the measures of performance s/he selected, in case the goals cannot be met simultaneously. This option activates the one variable heuristic if meeting both goals simultaneously is not possible, and the user is interested in meeting only one of the goals. If the user is not interested in meeting the goal for only one of the measures of performance, s/he is informed that goals cannot be reached.

The last form the user sees is the **Results** form (Figure 10), where the user is given the number of resources to reach the goals, the average value of the measures of performance selected and the number of iterations it takes to reach the goals.

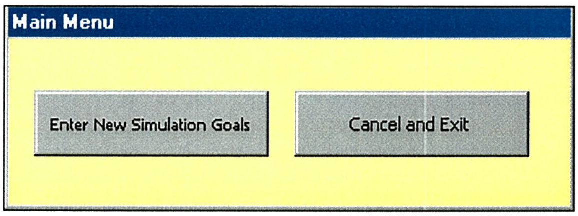


Figure 7: Main Menu Form

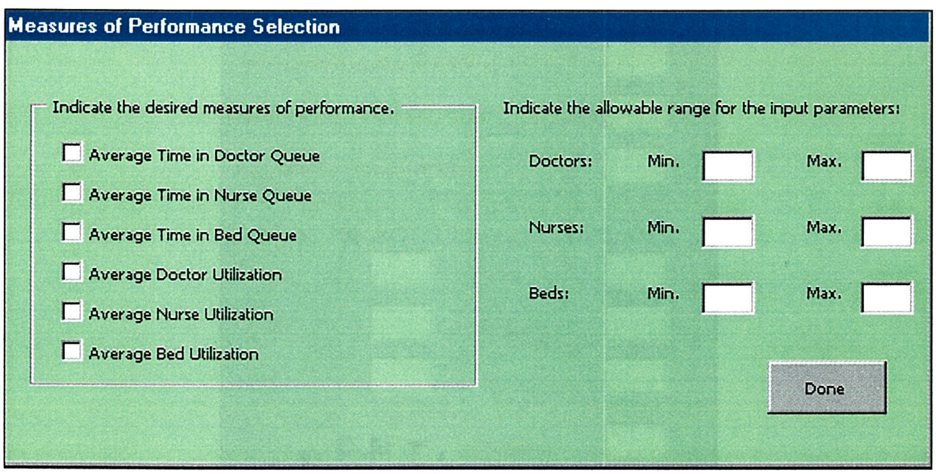


Figure 8: Performance Measure and Input Parameter Form

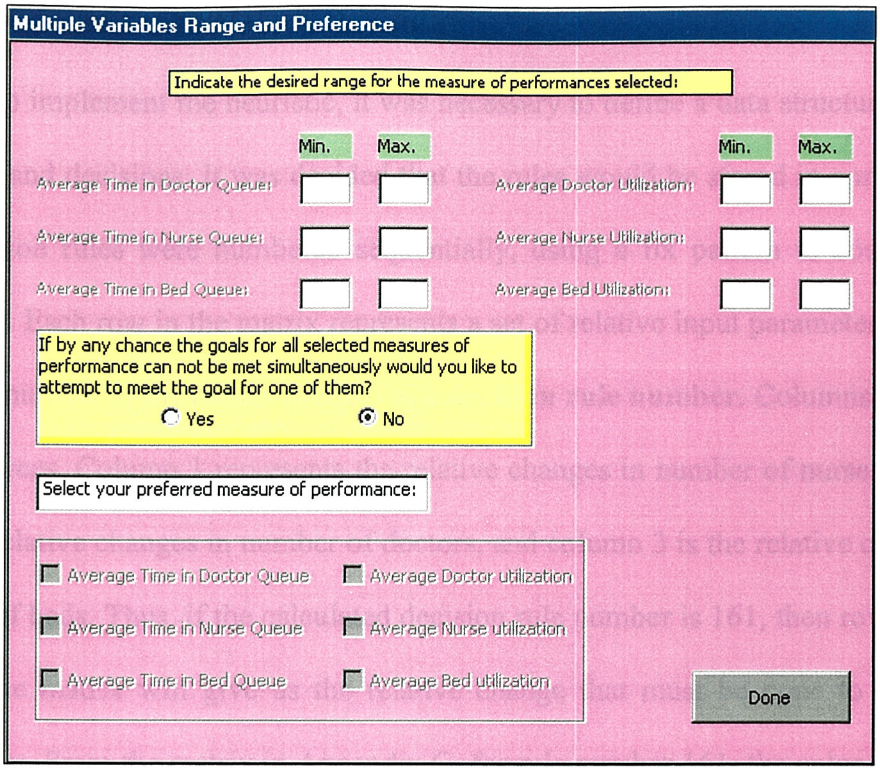


Figure 9: The Form for Capturing The Range of the Goals

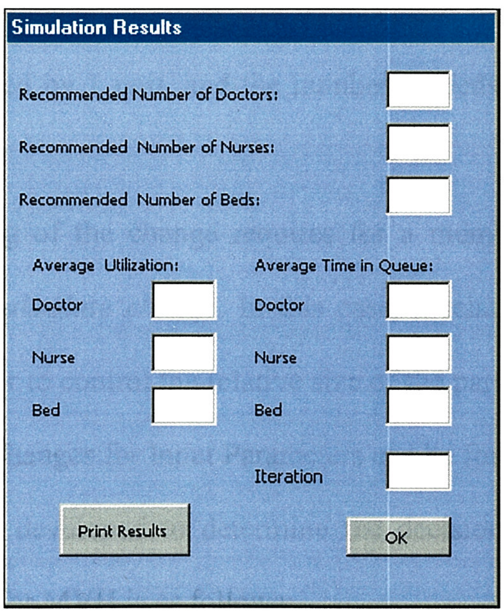


Figure 10: Simulation Results Form

5.2 Implementation of the Heuristic

To implement the heuristic, it was necessary to define a data structure to store the rules and decisions. It was decided that the rules would be stored in a matrix. All the decision rules were numbered sequentially, using a fix pattern to combine the scenarios. Each row in the matrix represents a set of relative input parameter changes. The row number in the matrix is equal to a **decision rule number**. Columns represent the resources. Column 1 represents the relative changes in number of nurses, column 2 is the relative changes in number of doctors, and column 3 is the relative changes in number of beds. Thus, if the calculated decision rule number is 161, then row number 161 in the matrix will give us the relative change that must be done to the input parameters. From the matrix in Appendix G, for rule number 161, the columns in that row give the vector,

$$(2 \quad 1 \quad 0),$$

which means that the number of nurses will be increased by 2 units, the number of doctors will be increased by 1 unit, and the number of beds will be increased by 0 units.

The representing of the change requires for a member of P to be either a *relative change* or an *absolute change*. In this case, a relative change was chosen because it made it easier to control the relative size of the parameter in the simulation model. The Matrix of Changes for Input Parameters can be found in Appendix G.

A formula was developed to determine the decision rule number for each scenario. This formula for MVH is as follows:

$$\textit{Decision Rule \#} = (A - 1) * 6 + B + (C-1)*36$$

Where:

A = Scenario Number of Goal 1 (A = 1, 2, ...6) (Table 11)

B = Scenario Number of Goal 2 (B = 1, 2, ...6) (Table 11)

C = Number of the Combination (C = 1, 2, ...15) (Table 12)

For example, Combination #8 is W_{qMD} and ρ_{beds}. So, if W_{qMD} is in Scenario #4 and ρ_{beds} is in Scenario #3, the decision rule to use should be 273 according to Table 38.

A = 4 B = 3 C = 8

Decision Rule # = (4-1)*6 + 3 + (8-1)*36 = 273

The number 6 in the equation is due to the fact that there are 6 scenarios for each variable considered. The number 36 is due to the combinations for any two variables. One important issue with this equation is that the number of the combination and the Scenario number for the goal should be carefully arranged; otherwise, it could lead to a mismatch. For this effort the order used is given in Table 11 and Table 15.

Table 15: Six measure of performance in order

Order #	Goal
1	Wq_{RN}
2	Wq_{MD}
3	Wq_{bed}
4	ρ_{RN}
5	ρ_{MD}
6	ρ_{bed}

As seen from Table 31 through Table 42, in some scenarios the decision is to use the one variable heuristic. In those cases, the rules in Table 46 through Table 51 are used.

The formula for the OVH is:

Decision Rule # $= A + (D\text{-}1)*6$

Where;

A = Scenario Number for the goal

D = Order number of the measure of performance

Implementing the heuristic using VBA required the development of several subroutines. These subroutines are called at various points during the Simulation Execution. Figure 11 shows the VBA Events and Subroutines necessary to implement *on-line* GDS.

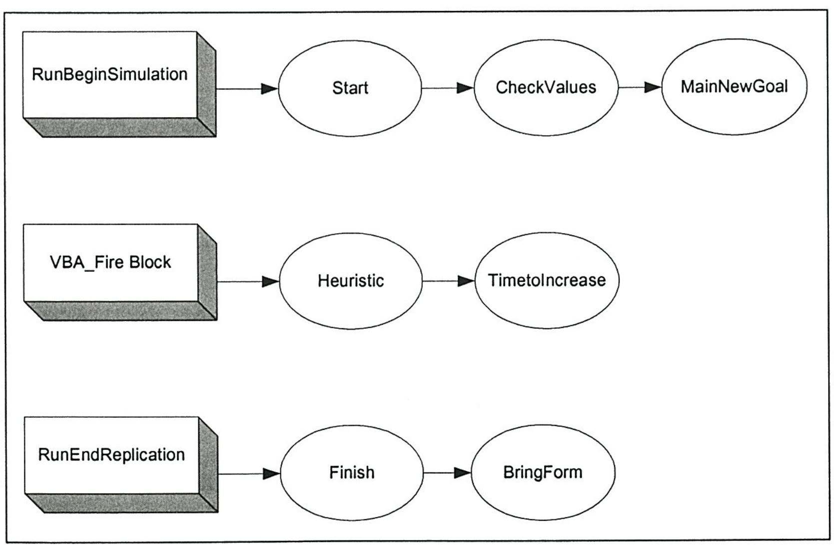


Figure 11:VBA Events and SubRoutines

At the beginning of the simulation run, the **RunBeginSimulation Event** triggers a routine to collect inputs from the user as well as updating the number of iterations in each stopping and restarting of the simulation. **CheckValues** checks if the values typed in the forms are correct. After that the **MainNewGoal** Subroutine is called to store all the values in the forms in the ARENA model, so that they can be accessed every time the simulation is stopped and/or input parameters need to be modified.

At each stoppage point, the **VBA_Fire block Event** is triggered to check the simulation outputs up to that point and to decide whether to change **P** or not. When **VBA_Fire block** is triggered it calls the **Heuristic** Subroutine, which is in charge of fetching outputs from ARENA, computing the G_i and 90% CI on the average of M_i and checking if the **P** can be increased, so that the decision of whether the model should continue to run or not can be made. The rules to change **P** are stored in a data file ("CMFILE.dat") for MVH and ("CM1FILE.dat") for OVH, which is then transferred to arrays.

If there is a need to change and the P can be changed **Time to Increase** Subroutine is called. This subroutine updates the Resource Module in the ARENA model.

When the run reaches the end of the replication, the **RunEndReplication** Event is triggered to report the results. After several iterations, the simulation run will stop either because it reached the end of the replication (simulation length) by meeting the goal, or because it cannot reach the goal. If the goal is met, the **Finish** and **Bring** Form Subroutines show the **Results** form displaying the recommended number of resources needed to accomplish the goals and the number of iterations it takes to reach them. If one of the goal scenarios is not feasible, the program checks if the user is interested in meeting only one of the goal. If the user is not interested, s/he is informed that the goal cannot be met, and the simulation ends.

5.3 Experiment Set 1: Testing of The Heuristic

After the heuristic was developed in VBA, it was important to test it to see if it was correct. Experiments were done to compare the results from the simulation run without the heuristic with the results from the simulation run with the heuristic.

The hypothesis tested is:

$H_0 : \varphi_M = \varphi_H$

$H_1 : \varphi_M \neq \varphi_H$

Where,

φ_M = Results from experiments without heuristic

φ_H = Results from experiments with heuristic.

The *t*-test assesses whether the means of two groups are statistically different from each other. This analysis is appropriate whenever we desire to compare the means of two groups. The comparison of the results is based on the configuration that meets the goals in both approaches, so a traditional *t*-test could not be done. Instead, it was checked if *at least one* of the configurations that meet the goals without the heuristic was same as the configuration that meets the goal with the heuristic. The heuristic at this moment only offers one configuration, but there may be several configurations that meet the same goal. The heuristic is not implemented to give all the possible configurations that meet the goal; this is left for further investigation.

Five different experiments for 3 different scenarios were done. The scenarios tested were:

1. Goals Met (GM)
2. Goals Not Met (GNM)
3. One of the Goals Met (OGM)

Table 16 gives the inputs for Experiment Set 1 and Table 17 gives the comparison of the results. After doing the experimentation, it was found that the configuration that met the goals with the heuristic is indeed **one** of the configurations that met the goals without the heuristic.

Table 16: Inputs for Experiment Set 1

Scenario	Exp	Goal1	Goal2	IPR			UGR1	UGR2
				RN	MD	Bed		
GM	1	ρ_{MD}	W_{qRN}	3-10	1-4	20-30	0.73-0.75	0.35-0.45
	2	ρ_{MD}	W_{qRN}	5-13	1-4	20-30	0.67-0.68	0.2-0.3
	3	ρ_{MD}	W_{qRN}	3-13	1-4	30-40	0.79-0.85	0.38-0.42
	4	ρ_{MD}	W_{qRN}	5-13	1-4	30-40	0.59-0.65	0.55-0.62
	5	ρ_{MD}	W_{qRN}	3-9	1-4	10-20	0.69-0.73	0.15-0.25
GNM	1	ρ_{MD}	W_{qRN}	3-8	2-4	20-30	0.91-0.94	0.45-0.55
	2	ρ_{MD}	W_{qRN}	5-9	2-4	20-30	0.78-0.83	0.15-0.18
	3	ρ_{MD}	W_{qRN}	3-9	2-4	30-40	0.82-0.85	0.21-0.25
	4	ρ_{MD}	W_{qRN}	3-8	2-4	20-30	0.85-0.88	0.27-0.28
	5	ρ_{MD}	W_{qRN}	3-9	2-4	30-40	0.79-0.83	0.14-0.16
OGM	1	ρ_{MD}	W_{qRN}	5-13	2-4	20-30	0.85-0.88	0.25-0.29
	2	ρ_{MD}	W_{qRN}	7-11	2-4	20-30	0.75-0.79	0.59-0.61
	3	ρ_{MD}	W_{qRN}	6-13	2-4	30-40	0.81-0.84	0.65-0.68
	4	ρ_{MD}	W_{qRN}	5-10	1-2	20-30	0.92-0.95	0.55-0.59
	5	ρ_{MD}	W_{qRN}	3-9	1-2	10-20	0.85-0.92	0.3-0.4

Table 17: Results from Experiment Set 1

Scenario	Exp.	Without Heuristic					With Heuristic				
		Configuration			Output		Configuration			Output	
		RN	MD	Bed	Goal1	Goal2	RN	MD	Bed	Goal1	Goal2
GM	1	9-13	1	20-30	≥0.83	<0.45	9	1	26	0.85	0.41
	2	8-13	1	20-30	≥0.77	<0.30	9	1	23	0.86	0.26
	3	10-13	1	30-40	≥0.77	<0.42	13	1	36	0.94	0.31
	4	10-13	1	30-35	≥0.82	<0.62	11	1	36	0.90	0.55
		9-13	1	35-40							
	5	6-9	1	10-15	≥0.68	<0.25	8	1	16	0.78	0.11
		9	1	15-20							
GNM	1	3-8	2-4	20-30	<0.80	>0.55	-	-	-	0.54	1.1
	2	5-9	2-4	20-30	<0.85	>0.18	-	-	-	0.76	0.56
	3	3-9	2-4	30-40	<0.65	>0.25	-	-	-	0.57	2.36
	4	3-8	2-4	20-30	<0.72	>0.28	-	-	-	0.45	1.78
	5	3-9	2-4	30-40	<0.68	>0.16	-	-	-	0.57	2.36
OGM	1	9-13	2-4	20-30	<0.73	<0.29	11	2	29	0.62	0.26
		10-13	2-4	30							
	2	8-13	2-4	20-28	<0.86	<0.61	11	2	30	0.62	0.30
		10-13	2-4	28-30							
	3	9-13	2-4	30-40	<0.75	<0.68	11	2	39	0.65	0.64
	4	7-10	1	20-28	<0.93	<0.59	9	1	28	0.84	0.56
		10	1	28-30							
		8-10	2	20-28							
		9-10	2	28-30							
	5	5-9	1	10-20	<0.95	<0.40	7	1	18	0.79	0.36
		7-9	2	10-20							

5.4 Experiment Set 2: On-line GDS versus At-End GDS

Experiment Set 2 was done to determine whether *on-line* GDS is better than *at-end* GDS. The factors in the criteria to decide on *better* were execution time, number of iterations required to meet the goal, and accuracy of the resulting configuration. Appropriate *t*-tests were done to compare the mean execution time, mean number of iterations, and mean accuracy under both approaches.

The hypotheses tested were:

1. $H_0 : \mu_{AE} \leq \mu_{OL}$ $\qquad H_1 : \mu_{AE} > \mu_{OL}$

2. $H_0 : \delta_{AE} = \delta_{OL}$ $\qquad H_1 : \delta_{AE} \neq \delta_{OL}$

3. $H_0 : \partial_{AE} = \partial_{OL}$ $\qquad H_1 : \partial_{AE} \neq \partial_{OL}$

4. $H_0 : \gamma_{AE} = \gamma_{OL}$ $\qquad H_1 : \gamma_{AE} \neq \gamma_{OL}$

Where,

μ_{AE} = Average execution time of MVH GDS heuristic using *at–end* approach.

μ_{OL} = Average execution time of MVH GDS heuristic using *on-line* approach.

δ_{AE} = Average number of iterations of MVH GDS heuristic using *at–end* approach.

δ_{OL} = Average number of iterations of MVH GDS heuristic using *on-line* approach.

∂_{AE} = Average output accuracy of MVH GDS heuristic using *at–end* approach.

∂_{OL} = Average output accuracy of MVH GDS heuristic using *on-line* approach.

γ_{AE} = Average midpoint accuracy of MVH GDS heuristic using *at–end* approach.

γ_{OL} = Average midpoint accuracy of MVH GDS heuristic using *on-line* approach.

Table 18 gives the inputs for Experiment Set 2, and Table 19 gives the results of these experiments. Table 20 shows the average execution times and number of iterations for both approaches.

Table 18: Inputs for Experiment Set 2

Scenario	Exp	Goal 1	Goal 2	IPR			UGR 1	UGR 2
				RN	MD	Bed		
GM	1	ρ_{MD}	W_{qRN}	5-10	1-4	20-30	0.73-0.75	0.52-0.62
	2	ρ_{MD}	W_{qRN}	4-10	1-4	20-30	0.67-0.68	0.39-0.41
	3	ρ_{MD}	W_{qRN}	4-13	1-4	30-40	0.79-0.85	0.52-0.57
	4	ρ_{MD}	W_{qRN}	4-10	1-4	30-40	0.59-0.65	0.92-0.95
	5	ρ_{MD}	W_{qRN}	4-9	1-4	10-20	0.69-0.73	0.35-0.43
GNM	1	ρ_{MD}	W_{qRN}	3-7	2-4	20-30	0.91-0.94	0.25-0.27
	2	ρ_{MD}	W_{qRN}	4-8	2-4	20-30	0.78-0.83	0.82-0.85
	3	ρ_{MD}	W_{qRN}	3-9	2-4	30-40	0.82-0.85	0.55-0.59
	4	ρ_{MD}	W_{qRN}	4-8	2-4	20-30	0.85-0.88	0.55-0.58
	5	ρ_{MD}	W_{qRN}	7-11	2-4	30-40	0.79-0.83	0.31-0.33
OGM	1	ρ_{MD}	W_{qRN}	6-12	2-4	20-30	0.85-0.88	0.92-0.95
	2	ρ_{MD}	W_{qRN}	4-8	2-4	20-30	0.75-0.79	1.8-2.1
	3	ρ_{MD}	W_{qRN}	8-13	2-4	30-40	0.81-0.84	0.80-0.85
	4	ρ_{MD}	W_{qRN}	5-9	1-4	20-30	0.92-0.95	0.61-0.65
	5	ρ_{MD}	W_{qRN}	4-8	1-4	10-20	0.85-0.92	0.52-0.54

Table 19: Results from Experiment Set 2

Scenario	Exp	At-End					On-Line				
		Configuration			Output		Configuration			Output	
		RN	MD	Bed	Goal1	Goal2	RN	MD	Bed	Goal1	Goal2
GM	1	7	1	23	0.82	0.60	9	1	23	0.86	0.26
	2	8	1	20	0.79	0.32	8	1	20	0.79	0.32
	3	12	1	37	0.92	0.44	12	1	33	0.91	0.34
	4	8	1	30	0.81	0.87	9	1	30	0.87	0.55
	5	6	1	15	0.76	0.39	7	1	17	0.78	0.28
GNM	1	7	2	26	0.45	1.78	7	2	26	0.45	1.78
	2	8	2	26	0.51	1.18	8	2	26	0.51	1.18
	3	9	2	39	0.57	2.36	9	2	39	0.57	2.36
	4	8	2	26	0.51	1.18	8	2	38	0.52	1.46
	5	11	2	36	0.64	0.51	11	2	36	0.64	0.51
OGM	1	10	2	30	0.6	0.65	10	2	30	0.60	0.65
	2	8	2	30	0.52	1.76	8	2	30	0.52	1.76
	3	11	2	40	0.64	0.63	11	2	40	0.64	0.63
	4	8	1	28	0.85	0.63	9	1	30	0.87	0.55
	5	7	1	19	0.77	0.44	8	1	20	0.79	0.32
	Average				**.6773**	**.9160**				**.6953**	**.8440**
	STD				**.1482**	**.6135**				**.1563**	**.6711**

Table 20: Average Execution Time and Number of Iterations for both approaches

Scenario	Experiment	Average execution time (min)		Number of Iterations	
		At-End	On-Line	At-End	On-Line
GM	1	12.44	7.49	2	3
	2	29.25	8.16	3	3
	3	38.16	12.07	6	5
	4	19.37	8.53	3	4
	5	24.42	12.17	4	5
GNM	1	20.35	2.51	3	3
	2	17.37	2.55	3	3
	3	19.31	3.34	4	4
	4	14	2.45	3	3
	5	10.42	2.57	3	3
OGM	1	20.12	4.52	5	5
	2	25.47	4.37	5	5
	3	16.53	4.54	5	5
	4	22.11	5.02	4	5
	5	41.19	7.37	6	7
	Average	**22.03**	**5.84**	**3.93**	**4.20**
	STD	**8.71**	**3.31**	**1.22**	**1.21**

Table 21: *t*-test for average execution time

Independent Samples Test

		t-test for Equality of Means						
							95% Confidence Interval of the Difference	
		t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
EXECTIME	Equal variances assumed	6.733	28	.000	16.19000	2.40465	11.26430	21.11570
	Equal variances not assumed	6.733	17.959	.000	16.19000	2.40465	11.13720	21.24280

The results of the *t*-test are summarized in Table 21. $t = 6.733 > t_{.05,28} = 2.048$. The *p*-value computed by SPSS is for a two-tailed test of significance. It must be adjusted for one tailed test. For this, the significance level (.000) has to be divided by 2 which is still .000 and **less than** .05; therefore the null hypothesis is rejected: The execution times are not the same. Now, looking at the 95% confidence interval, it is easy to conclude that $\mu_{AE} > \mu_{OL}$ because the *t*-test looked at the difference $\mu_{AE} - \mu_{OL}$; thus, the only way it can be > 0 is if $\mu_{AE} > \mu_{OL}$. Consequently, there is sufficient evidence to conclude that the mean execution time of the *on-line* multivariable GDS is **less than** the execution time of the *at-end* multivariable GDS.

Table 22: *t*-test for average number of iterations

Independent Samples Test

		t-test for Equality of Means						
							95% Confidence Interval of the Difference	
		t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
ITERAT	Equal variances assumed	-.601	28	.553	-.27	.44	-1.18	.64
	Equal variances not assumed	-.601	27.995	.553	-.27	.44	-1.18	.64

For the average number of iterations, the results of the two-sided *t*-test are summarized in Table 22. Since t = -.601 < $t_{.05,28}$ = 2.048 and the significance level (.553) is greater than .05, we fail to reject the null hypothesis. There is no sufficient evidence to conclude that the number of iterations for *on-line* multivariable GDS is different from *at-end* multivariable GDS. Therefore, OLMVH is no worse than AEMVH when it comes to the number of iterations.

To investigate if the two approaches gave different outputs, a t-test was performed on the outputs. The results of the *t*-test for the outputs obtained with both approaches are summarized in Table 23. The difference between first outputs is tested in Out1, and second output is tested in Out2.

For Output1 t = -.324< $t_{.05,28}$ = 2.048 and the significance level (.749) is greater than .05, we fail to reject the null hypothesis. There is no sufficient evidence to conclude that output1 obtained with *on-line* multivariable GDS is different from output1 obtained from *at-end* multivariable GDS. For Output2, t = .307< $t_{.05,28}$ = 2.048 and the significance level (.761) is greater than .05, we fail to reject the null hypothesis. There is no sufficient evidence to conclude that output2 obtained with *on-line* multivariable GDS is different from output2 obtained from *at-end* multivariable GDS. Again, OLMVH is no worse than AEMVH when it comes to the number of iterations.

Knowing that when the user enters a range for a goal, the value that s/he is typically thinking is the midpoint of the range, a *t*-test was done on the difference between the outputs and the midpoint of the goal range. The results of the *t*-test for the midpoint accuracy obtained with both approaches are summarized in Table 24.

Table 23: *t*-test on accuracy of Outputs

Independent Samples Test

		t-test for Equality of Means						
		t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference Lower	Upper
OUT1	Equal variances assumed	-.324	28	.749	-1.800E-02	5.5621E-02	-.13193	9.59E-02
	Equal variances not assumed	-.324	27.920	.749	-1.800E-02	5.5621E-02	-.13195	9.59E-02
OUT2	Equal variances assumed	.307	28	.761	7.2000E-02	.23479	-.40894	.55294
	Equal variances not assumed	.307	27.777	.761	7.2000E-02	.23479	-.40911	.55311

Table 24: *t*-test on midpoint accuracy

Independent Samples Test

		t-test for Equality of Means						
							95% Confidence Interval of the Difference	
		t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
MIDACU1	Equal variances assumed	-.247	28	.807	-1.800E-02	7.2997E-02	-.16753	.13153
	Equal variances not assumed	-.247	27.985	.807	-1.800E-02	7.2997E-02	-.16753	.13153
MIDACU2	Equal variances assumed	.305	28	.763	7.200E-02	.2362	-.4117	.5557
	Equal variances not assumed	.305	27.850	.763	7.200E-02	.2362	-.4119	.5559

The difference between first output and the midpoint of the goal range 1(UGR_1) is called MidAcu1, and the difference between second output and the midpoint of goal range 2 (UGR_2) is called MidAcu1. For MidAcu1 $t = -.247 < t_{.05,28} = 2.048$ and the significance level (.807) is greater than .05, we fail to reject the null hypothesis. There is no sufficient evidence to conclude that MidAcu1 obtained with *on-line* multivariable GDS is different from MidAcu1 obtained from *at-end* multivariable GDS. For MidAcu2, $t = .305 < t_{.05,28} = 2.048$ and the significance level (.763) is greater than .05, we fail to reject the null hypothesis. There is no sufficient evidence to conclude that MidAcu2 obtained with *on-line* multivariable GDS is different from MidAcu2 obtained from *at-end* multivariable GDS. Once more, OLMVH is no worse than AEMVH when it comes to the number of iterations.

Table 25 Gives a summary of results of Experiment Set 2. Based on these results, the use of multivariable *on-line* GDS yields about a **74% improvement** in execution time over multivariable *at-end* GDS.

Table 25: Results of the *t*-tests for Experiment Set 2

Null Hypothesis	Alternative Hypothesis	Results	Decision
$H_0 : \mu_{AE} \le \mu_{OL}$	$H_1 : \mu_{AE} > \mu_{OL}$	$t = 6.733 > t_{.05,28} = 2.048$	Reject H_0
$H_0 : \delta_{AE} = \delta_{OL}$	$H_1 : \delta_{AE} \ne \delta_{OL}$	$t = -.601 < t_{.05,28} = 2.048$	Fail to Reject H_0
$H_0 : \partial_{AE} = \partial_{OL}$	$H_1 : \partial_{AE} \ne \partial_{OL}$	$t = -.324 < t_{.05,28} = 2.048$ $t = .307 < t_{.05,28} = 2.048$	Fail to Reject H_0
$H_0 : \gamma_{AE} = \gamma_{OL}$	$H_1 : \gamma_{AE} \ne \gamma_{OL}$	$t = -.247 < t_{.05,28} = 2.048$ $t = .305 < t_{.05,28} = 2.048$	Fail to Reject H_0

CHAPTER 6

CONCLUSIONS AND FURTHER WORK

6.1 Conclusion of the Results

The goal of this investigation was to develop a methodology to extend the on-line GDS heuristic developed by Reyes (1998) to include two variables for non-terminating systems. Five objectives were proposed to meet this goal (Section1.2). In accordance, Reyes' ER model was enhanced and implemented using the newer version of the ARENA Software. Response Surface Analysis was done to see how changes in input parameters affected the measures of performance. From this analysis, rules to modify input parameters were derived and implemented using VBA for ARENA. Two sets of experiments were performed. Experiment Set 1 was done to verify correctness of the heuristic, whereas Experiment Set 2 was done to compare *on-line* GDS to *at-end* GDS.

The results of this research have proven that two variable *on-line* GDS is feasible and more efficient than *at-end* GDS. It was found that *on-line* GDS is significantly better in terms of execution time than *at-end* GDS, without affecting accuracy.

This research began with four basic questions (Section 1.1):

1. Can the goal set (Γ) for all variables be met?

2. Which variable should be favored in case Γ is not feasible?
3. Which input parameters should be changed to direct simulation results towards the goal set Γ?
4. By how much should the selected input parameters be changed?

This investigation has mainly contributed as follows in regards to these questions:

Can the goal set (Γ) for all variables be met? : Meeting both goals simultaneously may not be possible in the real world. Hence, the heuristic (MVH) has been designed to utilize Reyes' OVH to meet the goal of at least one of the variables.

What variable to favor in case Γ is infeasible? : Multi Attribute Utility Theory was tried to decide which variable to favor. But it was not successful because there was not enough information to determine the weight of each measures of performance. Since the MVH needed to work with at least one variable (using Reyes' OVH) the decision was made to ask the user which variable to favor. Thus in regards to this question the main contribution of this effort is that the heuristic does have a decision mechanism to either determine the preference using some heuristic (yet to be determined) or asking the user for such preference.

What input parameters to change to direct simulation results towards the goal set Γ? : The polynomials obtained from regression analysis were used to find which input parameters should be modified to direct the model towards the goal set Γ. It was found that in some cases multiple variable change would lead to a faster achievement of the goals. 540 general decision rules were derived. These rules could theoretically be used in other distinct, yet similar systems.

By how much to change the selected input parameters? : The percent effects of the input parameters were calculated from the polynomials to make a decision on how much to change the selected input parameter or parameters. A matrix of relative changes has been developed. The development of this matrix makes the MVH more efficient than Reyes' OVH, which used exhaustive search to modify input parameters. The use of this matrix leads to a faster convergence.

Summarizing, the following conclusions and deliverables have been obtained from this effort:

1. It is feasible to develop decision rules, from the response surface, to guide the simulation towards achieving a user's defined goal.
2. 540 general and portable decision rules to change input parameters have been developed.
3. Multiple resource changes lead to a faster achievement of a desired goal.
4. The heuristic allows triggering the OVH where appropriate. It also allows for the incorporation of a mechanism, to give preference to one of the measures of performance.
5. The *on-line* GDS MVH performs significantly better than *at-end* GDS MVH, yielding an average improvement of 74% in execution time.
6. It was confirmed that nurses are the most crucial resource for 4 of the 6 measures of performance. Correa (1999) had reached the same conclusion using a smaller sample set. Further beds are the least crucial for 5 of the 6

measures of performance. Even though beds do not affect the measures of performance very much, they seem to act as an adjusting factor.

7. Based on the work done by Jones (1999), it is reasonable to suggest that this work may be extended to terminating systems with minor adjustments.

It is worth noting that these results were obtained using two variables only. Extending these results to three or more variables should be relatively straightforward. Since the effect of single and multiple resource changes on each of the measure of performance has already been calculated, finding the decision rule for three variables is also possible. So when three variables out of six, there will be 216 decision rules for each combination and there will be a total of 20 different combinations. In the case of three variables instead of having 540 decision rules, there will be 4320 rules.

Obviously this is labor intensive. There is a big difference between deriving 540 decision rules and deriving 4320 decision rules. However, since the rules are based on the percent effects of each factor in the polynomials, instead of deriving the rules manually, they can be generated automatically. An algorithm that uses the coefficients of the polynomials in conjunction with vectors and matrices may be developed, so that the relative change matrix is generated dynamically. Once this algorithm is in place the cost of adding more variables to the MVH is relatively non-existent.

6.2 Limitations

Although a lot was learned from the effort, there are some limitations in the application and extensibility of the results.

- Given the time limitation, the goals for only two measures of performances can be achieved.
- The program implemented in VBA for ARENA for *on-line* GDS gives only one solution out of all possible configurations that would meet the goals.
- The program was implemented to give only one possible solution, but there may be many feasible configurations. The MVH does not find the best possible configuration.

6.3 Possible Extensions of this work

This research effort has shed light on some of the critical questions for GDS; however, it has also shown that there is still much to be done to achieve a true multivariable *on-line* GDS. There are several things that this research did not address, and that may be worth examining:

- *Thoroughly validate the polynomials and consider non-linear polynomials:* The number of experiments done to obtain the first-degree fitted polynomials can be increased to see if the polynomials still fit the data. Also the polynomials should be programmed in a tool such as Excel, and given numerous independent variables and their results compared to those from simulation. A closer look at the analysis of the response surface may lead to a second or third order relationship between the dependent and independent variables. For example focusing on the summary of analysis from SPSS for W_{qMD} (Table 26), it is possible to see that $R^2 = .429$. Although it is low this value suggests that there is a linear relationship between the dependent variables (nurse, doctor, and bed) and independent variable W_{qMD}. From the coefficient table, we can see that the significance for nurse is .156 which is greater than 0.05. Figure 12 makes it more clear with its $R^2 = 0.04$ that there is no linear relationship between W_{qMD} and nurse.

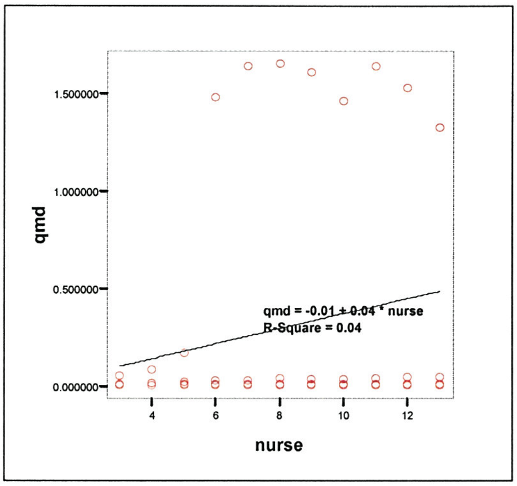


Figure 12: Regression line for W_{qMD} and number of nurses

Furthermore, the $F_0 = 25.020$ shows us that even though $\mathbf{F}_0 > \mathbf{F}_{.05;3;100} = 2.70$ this value is relatively low compared to the other F_0 values. All first-degree polynomials should be examined more thoroughly to evaluate the relationships and the possibilities of no-linear relationship.

Table 26: First degree fitted formula for W_{qMD}

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.655[a]	.429	.412	.46166572

a. Predictors: (Constant), NURSE, DOCTOR, BEDS

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	15.998	3	5.333	25.020	.000[a]
	Residual	21.314	100	.213		
	Total	37.312	103			

a. Predictors: (Constant), NURSE, DOCTOR, BEDS

b. Dependent Variable: WQMD

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.405	.225		1.799	.075
	BEDS	1.621E-02	.005	.271	3.592	.001
	DOCTOR	-.314	.040	-.586	-7.753	.000
	NURSE	2.791E-02	.020	.108	1.431	.156

a. Dependent Variable: WQMD

Figures 13 and 14 clearly show that a second-degree fitted polynomial will surely be a better fit for the data. The regression changes in Appendix D and the Summary of the Analysis of data should be examined together more thoroughly to find a better fit of the data.

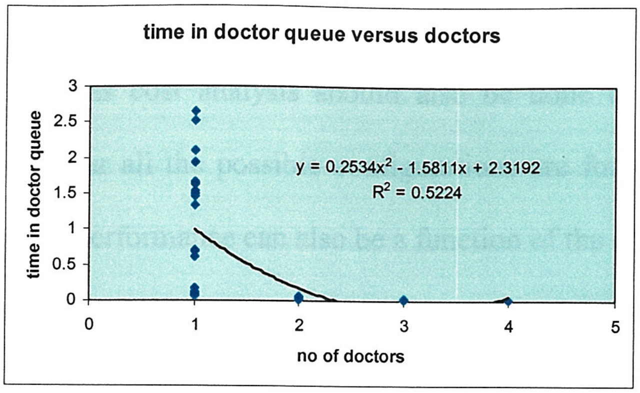


Figure 13: Second degree fitted polynomial for W_{qMD}

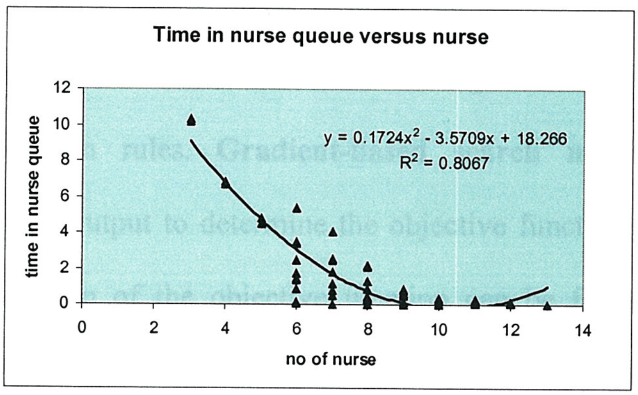


Figure 14: Second degree fitted polynomial for W_{qRN}

- *Examine the overlap:* In some of the scenarios, for example Scenario (6,3) for combination of W_{qMD} and ρ_{MD}, we do not know if it is more convenient to activate OVH or not. This is because we do not know exactly where the data lies in Scenario #3 ( ). It might be either close to the lower bound or the upper bound. The decision that should be done here may be a function of % overlap.
- *A recursive search:* The heuristic is not implemented to generate all the possible configurations. It could be programmed so that a recursive search is

done. Since increasing the number of nurses, doctors and beds is limited to the budget an ER has cost analysis should also be done to find an optimum configuration after all the possible configurations are found. The weights of the measures of performance can also be a function of the costs.

- *Handling conflicting goals:* The weights of the measures of performance can be changed depending on the number of resources and the measures of performance selected to decide what to favor when goals are conflicting.
- *Other Optimization Techniques:* Other optimization techniques such as gradient-based search methods and Genetic algorithms could also be used to develop decision rules. **Gradient-based search methods** estimate the gradient of the output to determine the objective function. With this method an approximation of the objective function can be found. Then from this approximation the decision rules can be derived. One draw back of this method is that it. often fails to find the global maximum of the objective function, only local maximum can be found (Carson and Maria, 1997). **Genetic algorithms** (GA) are inspired by Darwin's theory about evolution. The search procedure that GA is built upon finds a set of variables that optimizes the fitness of an individual and/or of the whole population. Since this method simultaneously searches in many directions, the probability of finding a global optimum greatly increases. Research using the GA for optimization has demonstrated its strong potential for obtaining globally optimal solutions (Goldberg, 1989). These methods are successfully used in

other research efforts to develop algorithms to drive simulation optimization and it is reasonable to suggest the use of those techniques in this system.

- *Automatic generation of the rules:* To reduce the set up cost of adding variables, it is necessary to develop an algorithm to automatically generate the decision rules and relative changes.

REFERENCES

Alsugair, A. and D. Y. Chang, 1994, "A Goal Driven Approach to Discrete Event Simulation," *Proceedings of the First Congress held in Conjunction with A/E/C Systems*, K. Khozeimeh (ed.), 512-522.

Baesler, F. F. and J. Sepulveda, 2000, "Multi-response Simulation Optimization Using Stochastic Genetic Search Within a Goal Programming Framework," *Proceedings of the 2000 Winter Simulation Conference*, J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, (eds.), 788-794.

Banks, J., 1999, "Introduction to Simulation," *Proceedings of the 1999 Winter Simulation Conference*, P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans (eds.), 7-13.

Boyle, C. R., 1996, "An Interactive Multiple Response Simulation Optimization Method," *IIE Transactions*, 28: 453-463.

Carson, Y. and A. Maria, 1997, "Simulation Optimization: Methods and Applications," *Proceedings of the 1992 Winter Simulation Conference*, J. A. Jones, R.R Barton, K. Kang, and P. A. Fischwick (eds.), Orlando, FL., 1978-1984.

Centeno, M. A. and C. R. Standridge, 1992, "Databases and Artificial Intelligence: Enabling Technologies for Simulation Modeling," *Proceedings of the 1992 Winter Simulation Conference*, J. J. Swain, D. Goldsman, R. C. Crain, and J. R. Wilson (eds.), 181-188.

Centeno, M. A. and E. R. Forrest, 1999, "Assessing the Portability of Goal Driven Simulation on a Manufacturing System", Report, Research Experience for Undergraduates (REU), Department of Industrial and Systems Engineering, Florida International University, Miami, FL.

Chen, S., L. Chen, and L. Lin, 2001, "Knowledge-based Support for Simulation Analysis of Manufacturing Cells," *Computers in Industry*, 44, 33-49.

Clayton, E. R., W. E. Weber, and B. W. Taylor, 1982, "A Goal Programming Approach to the Optimization of Multi-Response Simulation Models," *IIE Transactions*, 14: 282-287.

Correa, D., 1999, *A Study of Response Surface in Simulation of Emergency Room Systems*, Master's Thesis, Department of Industrial and Systems Engineering, Florida International University, Miami, FL 33199.

Cochran, W. G. and G. M. Cox, 1962, *Experimental Designs*, New York, John Wiley & Sons.

DeWispelare, A. R. and A. P. Sage, 1981, "On Combined Multiple Objective Optimization Theory and Multiple Attribute Utility Theory for Evaluation and Choice Making," *Large Scale Systems*, Vol:2, 1-19.

Edwards, W. and J. R. Newman, 1982, *Multi-Attribute Evaluation*, Sage Publications, Beverly Hills, CA.

Ford, D. R. and B. J. Schroer, 1987, "An Expert Manufacturing Simulation System," *Simulation*, 48:5, 193-200.

Fox, M. and Y. Reddy, 1982, "Knowledge Representation in Organizational Modeling and Simulation: Definition and Interpretation," *Proceedings of the 13th Annual Pittsburgh Conference on Modeling and Simulation*, 675-683.

Fu, M. C., 1994, "Optimization via Simulation: a Review," *Annals of Operations Research*, 53: 199-247.

Garcia, M. L., M. A. Centeno, C. Rivera, and N. DeCario, 1995, "Reducing Time in an Emergency Room Via a Fast Track," *Proceedings of the 1995 Winter Simulation Conference*, C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman (eds.), 1048-1053.

Goldberg, D.E., 1989, Genetic Algorithms in Search, Optimization and Machine Learning, Addison–Wesley, Reading, MA.

Jain, S., 1999, "Simulation in the Next Millennium," *Proceedings of the 1999 Winter Simulation Conference*, P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans (eds.), 1478-1484.

Jones, J. M., 1999, *On-Line Goal Driven Simulation For Terminating Systems,* Master's Thesis, Department of Industrial and Systems Engineering, Florida International University, Miami, FL 33199.

Klahr, P., W. S. Faught, and G. R. Martins, 1980, "Rule Oriented Simulation," *Proceedings of 1980 International Conference on Cybernetics and Society*, IEEE, Cambridge, Ma, 350-354.

Kleijnen, J. P. C., 1998, "Experimental Design for Sensitivity Analysis, Optimization, and Validation of Simulation Models," *Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice,* J. Banks (ed.), 172-223, New York: John Wiley & Sons.

Matwiczak, K. M., 1990, *AMOS – Automated Multiple-Response Optimum-Seeker for Simulations*, Ph.D. Dissertation, Department of Industrial Engineering, Texas A&M University, College Station, TX.

Mellichamp, J. M., O. Kwon, and A. F. A. Wahab, 1990, "FMS Designer: An Expert System for Flexible Manufacturing Systems Design," *International Journal of Production Research,* 28 (1), 2013-2024.

Molina, L., C. Gandarillas, and M. A. Centeno, 1996, "Goal Driven Simulation Intelligent Back Ends: A State of the Art Review," *Proceedings of the 1996 Winter Simulation Conference*, J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain (eds.), 734-739.

Mollaghasemi M. and G. W. Evans, 1993, "A Methodology for Stochastic Optimization of Multiple Response Simulation Models," *Engineering Optimization*, Vol. 21, 265-275.

Mollaghasemi, M. and G. W. Evans, 1994, "Multicriteria Design of Manufacturing Systems Through Simulation Optimization," *Transactions on Systems, Man, and Cybernetics*, 24: 1407-1411.

Mollaghasemi, M., 1994, "Multiple Response Optimization of Simulation Models," *Transactions of the Society for Computer Simulation*, 11: 179-192.

Mollaghasemi, M., G. W. Evans, and W. E. Biles, 1991, "An Approach for Optimizing Multiple Response Simulation Models," *Computers and Industrial Engineering*, 21, 201-203.

Montgomery, D. and V. Bettencourt, 1977, "Multiple Response Surface Methods in Computer Simulation," *Simulation*, 29:1407-1411.

Morrice, D. J., J. Butler, P. W. Mullarkey, 1998, "An Approach to Ranking and Selection for Multiple Performance Measures," *Proceedings of the 1998 Winter Simulation Conference*, D. J. Medeiros, E. F. Watson, J. S. Carson, and M. S. Manivannan (eds), 719-725.

Myers, R. H. and D. C. Montgomery, 1995, Response Surface Methodology: Process and Product Optimization Using Designed Experiments, New York: John Wiley & Sons.

Neddermeijer, G. H., G. J. van Oortmarssen, N. Piersma, and R. Dekker, 2000, "A Framework for Response Surface Methodology for Simulation Optimization," *Proceedings of the 2000 Winter Simulation Conference*, J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick (eds.), 788-794.

O'Keefe, R., 1986, "Simulation and Expert Systems- A Taxonomy and Some Examples," *Simulation,* 46 (1), 10-16.

Page, M., J. Gensel, and M. Boudis, 1999, "An Algorithm For Goal Driven Simulation," *Proceedings of the 1999 Winter Simulation Conference*, P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans (eds.), 578-585.

Penaloza, G. C., 1999, *A Forecasting Based Heuristic for On-line Goal Driven Simulation*, Master's Thesis, Department of Industrial and Systems Engineering, Florida International University, Miami, FL 33199.

Prakash S. and R. E. Shannon, 1989, "Intelligent Back-End of a Goal Directed Simulation Environment for Discrete Part Manufacturing." *Proceedings of the 1989 Winter Simulation Conference*, E. A. MacNair, K. J. Musselman, and P. Heidelberger (eds), 883-891.

Rees, L. P., E. R. Clayton, and B. W. Taylor, 1985, "Solving Multiple Response Simulation Optimization Methods Using Modified Response Surface Methodology Within a Lexicographic Goal Programming Framework," *IIE Transactions*, 17: 447-457.

Reyes, M. F., 1998, *A Heuristic for On-Line Assessment of Simulation Output for Goal Driven Simulation*, Master's Thesis, Department of Industrial and Systems Engineering, Florida International University, Miami, FL 33199.

Rothenberg, J., 1990, "Artificial Intelligence and Simulation (Tutorial)," *Proceedings of the 1990 Winter Simulation Conference*, O. Balci, R. P. Sadowski, R. E. Nance (eds.), 22-24.

Shannon, R. E., 1988 "Knowledge Based Simulation Techniques for Manufacturing", International Journal of Production Research, 3, 953-973.

Tamiz, M., D. Jones, and C. Romero, 1998, "Goal Programming for Decision Making: An Overview of the Current State-of-the-Art," *European Journal of Operational Research*, 111: 569-581.

Teleb R. and F. Azadivar, 1994, "A Methodology for Solving Multi-Objective Simulation-Optimization Problems," *European Journal of Operational Research*, 72: 135-145.

Touran, A., 1990, "Integration of Simulation with Expert Systems," *Journal of Construction Engineering and Management*, 116 (3), 480-493.

Von Neumann, J. and O. Morgenstern, 1947, *Theory of Games and Economic Behavior*, Second edition, Princeton University Press, Princeton, NJ.

Zionts, S., 1992, "Some Thoughts on Research in Multiple Criteria Decision Making," *Computers and Operations Research*, 19: 7, 567-570.

ADDITIONAL REFERENCES

Centeno M. A. and M. Carrillo, 2001, "Challenges of Introducing Simulation as a Decision Making Tool," *Proceedings of the 2001 Winter Simulation Conference*, B.A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer (eds.), 17-21.

Myers, R. H., 1976, *Response Surface Methodology*, Edwards Brothers, Ann Arbor. MI.

APPENDICES

APPENDIX A

A.1 Model Specifications

Table 27: Specifications for Stations

Stations	Explanation
MyEntrance	Begins the logic for the real entrance to the system
MyExit	Serves as the final stage of the simulation where final statistics are collected
registrationstation	Begins the logic for the registration process
TreatmentStation	Begins the logic for the treatment process
TriageStation	Begins the logic for the Triage Area
waitingroom	Begins the logic for the waiting area

Table 28: Specifications for Resources

Resources	Explanation
bed	Regular bed
extrabed	Extra bed
FT	Fast Track Bed
MD	Physician
Nurse	Registered Nurse
RegistrationRep	Registration Representative
triagenurse	Triage Nurse

Table 29: Specifications for Queues

Queues	Explanation
Dummy	Queue before the seizing of the beds
MD_Q	Queue for the Physician
NurseQ_1	Queue for the Initial Nurse visit
NurseQ_2	Queue for the Nurse visit after the physician.
NurseQ_3	Nurse Queue for the observation check up
RegRepQ	Queue for Registration Representative
Triage_Q	Queue of the patients waiting for the Triage Nurse
WaitQ_1	Patients Category 4 in the waiting room
WaitQ_2	Patients Category 3 in the waiting room
WaitQ_3	Patients Category 2 in the waiting room

Table 30: Specifications for Attributes

Attributes	Explanation
arriveby	To assign by what means the patient arrives
decision	The decision of the doctor: observed, admitted or discharged
doctortime	The time that the physician spends with the patients (differs by pcategory)
nursetime	The time that the RN spends with the patients (differs by pcategory)
observationtime	The total time that the patients are being observed in the ER (differs by pcategory)
pcategory	Patient category
Timein	To put a time stamp to the time that the patient arrives to the system
tnowenteringloop	To put a timestamp to the time that the patient enters the Observation area logic
whichbed	To assign the bed the patient will be using
TimeNurse1	To put a timestamp to the time the patient waits for nurse
TimeNurse2	To put a timestamp to the time the patient waits for nurse
TimeNurse3	To put a timestamp to the time the patient waits for nurse
timeMD	To put a timestamp to the time the patient waits for doctor
timeDummy	To put a timestamp to the time the patient waits in the dummy queue
timeWR	To put a timestamp to the time the patient spends in the waiting room

A.2 TSM Experiment Frame Text Version

```
ATTRIBUTES:
arriveby,0:
timeDummy,0:
nursetime,0:
tnowenteringloop,0:
timeWR,0:
pcategory,0:
Timein:
observationtime,0:
timeMD,0:
timeNurse1:
timenurse2:
timeNurse3:
doctortime,0:
decision,0:
whichbed,0;
```

```
QUEUES:
1,WaitQ1,FIFO,
2,WaitQ2,FirstInFirstOut
3,WaitQ3,FirstInFirstOut
4,nurseQ_1,LVF(pcategory)
5,NurseQ_2,LVF(pcategory)
6,NurseQ_3,LVF(pcategory)
7,triage_Q,LowValueFirst(pcategory)
8,MD_Q,LowValueFirst(pcategory)
9,Dummy,LVF(pcategory)
10,RegRepQ,LVF(pcategory)

STATIONS:
1,registrationstation:
2,waitingroom:
3,TreatmentStation:
4,TriageStation;
5,MyExit
6,MyEntrance
```

```
RESOURCES:
1,Nurse,Capacity(6),
2,MD,Capacity(1),
3,bed,Capacity(28),
4,extrabed,Capacity(1),
5,RegistrationRep,
6,FT,Capacity(1),
7,triagenurse,Capacity(2),
```

```
NICKNAMES:
1,ambulance,1:
2,owncar,2:
3,emergent,1:
4,urgent,2:
5,nonurgent,3:
6,stable,4:
7,observe,1:
8,admit,2
9,discharge,3;
```

```
DSTATS:
1,nq(waitQ1),# of people in waitQ1:
2,NR(Nurse),Nurse Busy:
3,NR(MD),MD Busy:
4,nr(bed),Bed Busy:
5,MR(MD),MD Available:
6,MR(Nurse),Nurse Available:
7,mr(bed),Beds Available:
8,nq(waitQ2),no of people in waitQ2:
9,nq(waitQ3),no of people in waitQ3:
10,nq(dummy),no of people in dummy q:
11,nq(MD_Q),no of people in doctor q:
12,nq(nurseQ_1),no in nurse Q1:
13,nq(nurseQ_2),no in nurse Q_2:
14,nq(nurseQ_3):
15,nr(nurse)/mr(nurse),util of nurse:
16,nr(md)/mr(md),md util:
17,nr(bed)/mr(bed),bed util:
18,NQ(RegRepQ),# in RegRepQ;
```

```
TALLIES:
1,TWaitQ1,"WAITq1.DAT":
2,TWaitQ2,"WAITQ2.DAT":
3,TWaitQ3,"WAITQ3.DAT":
4,TDummy,"Dummy.DAT":
5,Time ER 1,"ER11.dat":
6,Time ER 2,"ER12.dat":
7,Time ER 3,"ER13.dat":
8,Time ER 4,"ER14.dat":
9,Time in Bed 1,"bed1.dat":
10,Time in Bed 2,"bed2.dat":
11,Time in Bed 3,"bed3.dat":
12,Time in Bed 4,"bed4.dat":
13,TNurseQ_1:
14,TNurseQ_2:
15,TNurseQ_3:
16,TMD_Q:
RegRepQ Queue Time;
```

```
COUNTERS:
1,arrivebyambulance,
2,arrivebyowncar,
3,Patients balking,
4,people in waiting room,
5,Patients leaving,
REPLICATE,
1,0.0,5300,24.0,Hours,1000;
```

```
SETS:
1,BedsP1,Bed,extrabed:
2,BedsP2,Bed,extrabed:
3,BedsP3,FT,Bed,extrabed:
4,BedsP4,FT,Bed:
5,Time in Er, Time Er 1,
              Time Er 2,
              Time Er 3,
              Time Er 4;
```

```
EXPRESSIONS:
1,doctor1,uniform(0.17,0.33):
2,doctor2,uniform(0.167,0.25):
3,doctor3,uniform(0.0167,0.083):
4,doctor4,uniform(0.0167,0.083):
5,nurse1,uniform(0.5,1):
6,nurse2,uniform(.5,.67):
7,nurse3,uniform(.33,.67):
8,nurse4,uniform(.05,.5):
9,PatientTypeAmbulance,discrete(.435,1,.82,2,1,3):
10,PatientTypeOwnCar,Discrete(.026,1,.471,2,.967,3,1,4):
11,Observation1,max(0,erla(1.02,4)-doctortime-nursetime):
12,Observation2,Max(0,max(norm(3.84,2.05),1.79)-doctortime-nursetime):
13,Observation3,Max(0,gamm(1.4,1.5)-nursetime-doctortime):
14,Observation4,Max(0,max(norm(0.552,0.295),.257)-doctortime-nursetime)
15,admitted,Uniform(.35,.5)+uniform(1/60,4/60):
16,discharged,uniform(.35,.5);
```

```
VARIABLES:
1,TBA,CLEAR(System),CATEGORY("None-None"),0:
2,MinRes1Util,CLEAR(System),CATEGORY("None-None"):
3,MaxRes1Util,CLEAR(System),CATEGORY("None-None"),0:
4,MinRes2Util,CLEAR(System),CATEGORY("None-None"):
5,MaxRes2Util,CLEAR(System),CATEGORY("None-None"),0:
6,MinRes3Util,CLEAR(System),CATEGORY("None-None"),0.85:
7,MaxRes3Util,CLEAR(System),CATEGORY("None-None"),0.95:
8,MinTimeQ1,CLEAR(System),CATEGORY("None-None"):
9,MaxTimeQ1,CLEAR(System),CATEGORY("None-None"),0:
10,MinTimeQ2,CLEAR(System),CATEGORY("None-None"):
11,MaxTimeQ2,CLEAR(System),CATEGORY("None-None"),0:
12,MinTimeQ3,CLEAR(System),CATEGORY("None-None"),1.2:
13,MaxTimeQ3,CLEAR(System),CATEGORY("None-None"),1.3:
14,MeasureTINQ,CLEAR(System),CATEGORY("None-None"),0:
15,MeasureTIDQ,CLEAR(System),CATEGORY("None-None"),0:
16,MeasureTIBQ,CLEAR(System),CATEGORY("None-None"),1:
17,MeasureANU,CLEAR(System),CATEGORY("None-None"),0:
18,MeasureADU,CLEAR(System),CATEGORY("None-None"),0:
19,MeasureABU,CLEAR(System),CATEGORY("None-None"),1:
20,NurseFlag,CLEAR(System),CATEGORY("None-None"),0:
21,DocFlag,CLEAR(System),CATEGORY("None-None"),0:
22,BedsFlag,CLEAR(System),CATEGORY("None-None"),0:
23,MaxNur,CLEAR(System),CATEGORY("None-None"),10:
24,MaxDoc,CLEAR(System),CATEGORY("None-None"),4:
25,MaxBeds,CLEAR(System),CATEGORY("None-None"),30;
```

A.3 TSM Model Frame Text Version

```
0$              CREATE,         1,0.0:expo(0.125):NEXT(1$);
1$              ASSIGN: arriveby=discrete(0.0818,ambulance,1.0,owncar);
20$             COUNT:          ArriveBy,1;
8$              ASSIGN:         pcategory=ed(ArriveBy + 8):MARK(Timein);
62$             ROUTE:          0.0,MyEntrance;
5$              STATION,        waitingroom:MARK(timeWR);
15$             COUNT:          people in waiting room,1:
74$             BRANCH,         1:
                                If,pcategory==stable,24$,Yes:
                                If,pcategory==nonurgent,38$,Yes:
                                Else,23$,Yes;
24$             QUEUE,          waitQ1;
28$             SCAN: nr(ft)<mr(ft).or.nr(bed)<mr(bed):
75$             TALLY:          TWaitQ1,Interval(timeWR),1:NEXT(59$);
59$             COUNT:          p4_goingforbed,1;
35$             ROUTE:          .033,TreatmentStation;
38$             QUEUE,          waitQ2;
39$             SCAN:
nr(ft)<mr(ft).or.nr(bed).lt.mr(bed).or.nr(extrabed).lt.mr(extrabed):
77$             TALLY:          TWaitQ2,Interval(timeWR),1:NEXT(60$);
60$             COUNT:          p3_goingforbed,1:NEXT(35$);
23$             QUEUE,          waitQ3;
29$             SCAN: nr(bed)<mr(bed).or.nr(extrabed).lt.mr(extrabed):
79$             TALLY:          TWaitQ3,Interval(timeWR),1:NEXT(61$);
61$             COUNT:          p2_goingforbed,1:NEXT(35$);
9$              STATION,        TriageStation;
54$             QUEUE,          triage_Q;
55$             SEIZE,          1,Other:
```

```
                                    triagenurse,1:NEXT(10$);
84$             DELAY:              UNIF( .0833,.1667),,Other:NEXT(56$);
56$             RELEASE:            triagenurse,1:NEXT(11$);
11$             TRACE,              -1,"-Choosing from 2 options\n":;
85$             BRANCH,             1:
                                    If,pcategory==emergent,21$,Yes:
                                    Else,3$,Yes;
21$             ROUTE:              0.033,TreatmentStation;
3$              ROUTE:              0.033,registrationstation;
registration    STATION,            registrationstation:MARK(Timein);
18$             QUEUE,              RegRepQ:
89$             SEIZE,              ,Other:
                                    RegistrationRep,1:NEXT(94$);
94$             ASSIGN:             j=j;
90$             TALLY:              RegRepQ Queue Time,INT(QueueTime),1:NEXT(12$);
95$             DELAY:              EXPO( .1667),,Other:NEXT(13$);
96$             RELEASE:            RegistrationRep,1:NEXT(17$);
17$             ROUTE:              0.033,waitingroom;
37$             STATION,            TreatmentStation;
57$             QUEUE,              Dummy:MARK(timeDummy);
36$             SEIZE,              1,Other:
Select(Pcategory,POR,WhichBed),1:NEXT(69$);
97$             TALLY:              TDummy,Interval(timeDummy),1:NEXT(41$);
41$             ASSIGN:             nursetime=ed(pcategory+4):
                                    doctortime=ed(pcategory);
43$             QUEUE,              nurseQ_1:MARK(timeNurse1);
42$             SEIZE,              1,Other:
                                    Nurse,1:NEXT(70$);
99$             TALLY:          TNurseQ_1,Interval(timeNurse1),1:NEXT(14$);
101$            DELAY:              nursetime*1.5,,Other:NEXT(44$);
44$             RELEASE:            Nurse,1;
46$             QUEUE,              MD_Q:MARK(timeMD);
45$             SEIZE,              1,Other:
                                    MD,1:NEXT(71$);
102$            TALLY:              TMD_Q,Interval(timeMD),1:NEXT(16$);
104$            DELAY:              doctortime,,Other:NEXT(47$);
47$             RELEASE:            MD,1;
49$             QUEUE,              NurseQ_2:MARK(timenurse2);
48$             SEIZE,              1,Other:
                                    Nurse,1:NEXT(72$);
105$            TALLY:          TNurseQ_2,Interval(timenurse2),1:NEXT(34$);
107$            DELAY:              NurseTime * .25,,Other:NEXT(50$);
50$             RELEASE:            Nurse,1;
32$             ASSIGN:             decision=discrete(0.5,1,.75,2,1,3);
30$             BRANCH,             1,10:
                                    If,decision==observe,33$,Yes:
                                    Else,31$,Yes;
33$             ASSIGN:
observationtime=ed(pcategory+10):MARK(tnowenteringloop):NEXT(27$);
108$            DELAY:              uniform(.35,.5),,Other:NEXT(52$);
52$             QUEUE,              NurseQ_3:MARK(timeNurse3);
51$             SEIZE,              1,Other:
                                    Nurse,1:NEXT(73$);
109$            TALLY:          TNurseQ_3,Interval(timeNurse3),1:NEXT(25$);
111$            DELAY:              uniform(1/60,4/60),,Other:NEXT(53$);
53$             RELEASE:            Nurse,1;
26$             BRANCH,             1,10:
If,tnow.ge.observationtime+tnowenteringloop,64$,Yes:
                                    Else,27$,Yes;
64$             ROUTE:              0.0,MyExit;
31$             DELAY:              ed(decision+13),,Other:NEXT(64$);
```

```
63$             STATION,        MyEntrance;
22$             BRANCH,         1,10:
                                If,ArriveBy .eq. Ambulance,2$,Yes:
                                Else,4$,Yes;
2$              BRANCH,         1,10:
If,nr(bed).lt.mr(bed).or.nr(extrabed).lt.mr(extrabed),go_bed,Yes:
                                Else,NoBeds,Yes;
go_bed          ROUTE:          .033,TreatmentStation;
112$            COUNT:          Patients balking,1:NEXT(58$);
58$             DISPOSE:        No;
4$              ROUTE:          0.033,TriageStation;
65$             STATION,        MyExit;
6$              RELEASE:        member(pcategory,whichbed),1:NEXT(40$);
115$            TALLY:          Time in
Er(pcategory),Interval(Timein),1:NEXT(7$);
7$              COUNT:          Patients leaving,1;
bye             DISPOSE:        No;
```

A.4 TSM Model Graphical Version

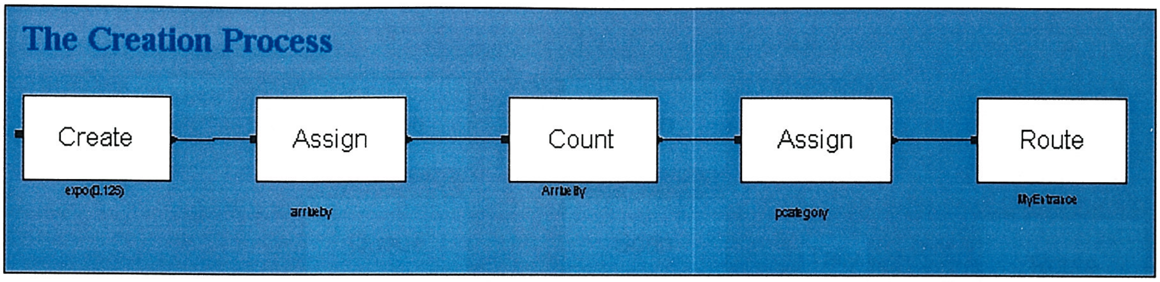


Figure 15: The Creation Process

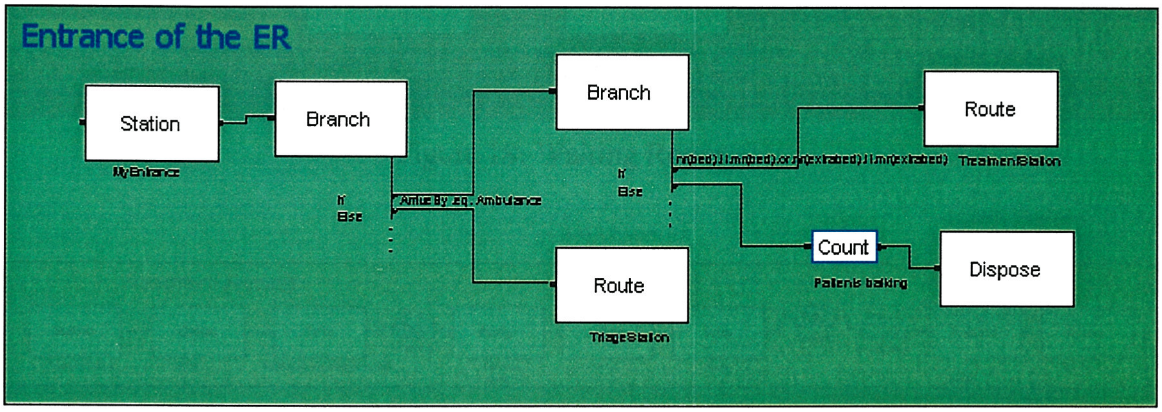


Figure 16: Entrance to the ER

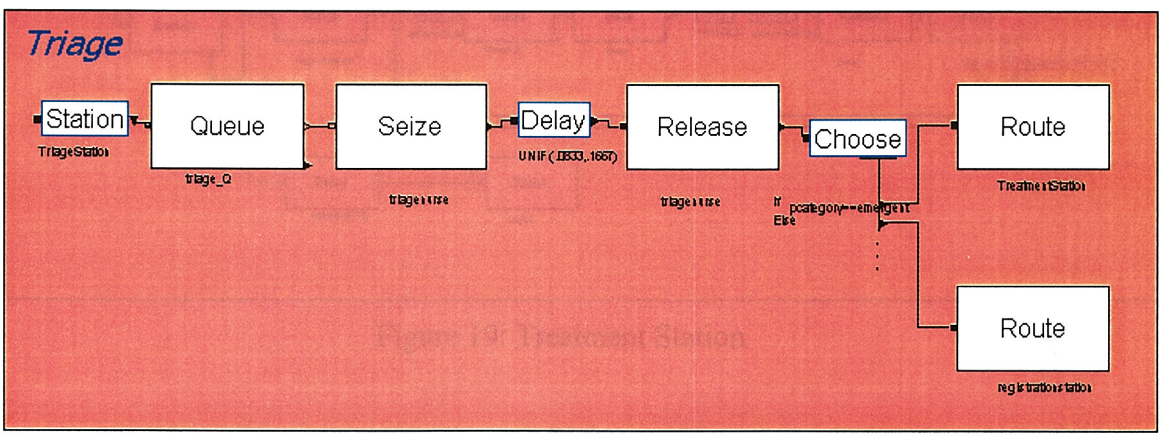


Figure 17: Triage Area

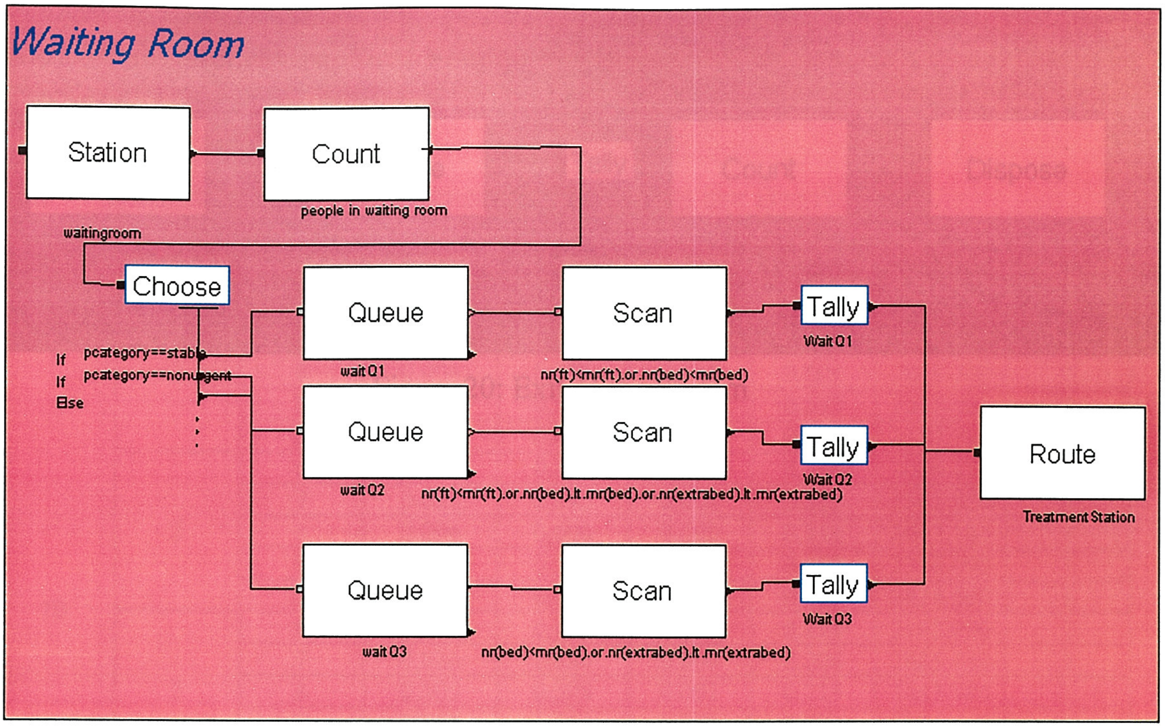


Figure 18: Waiting Room

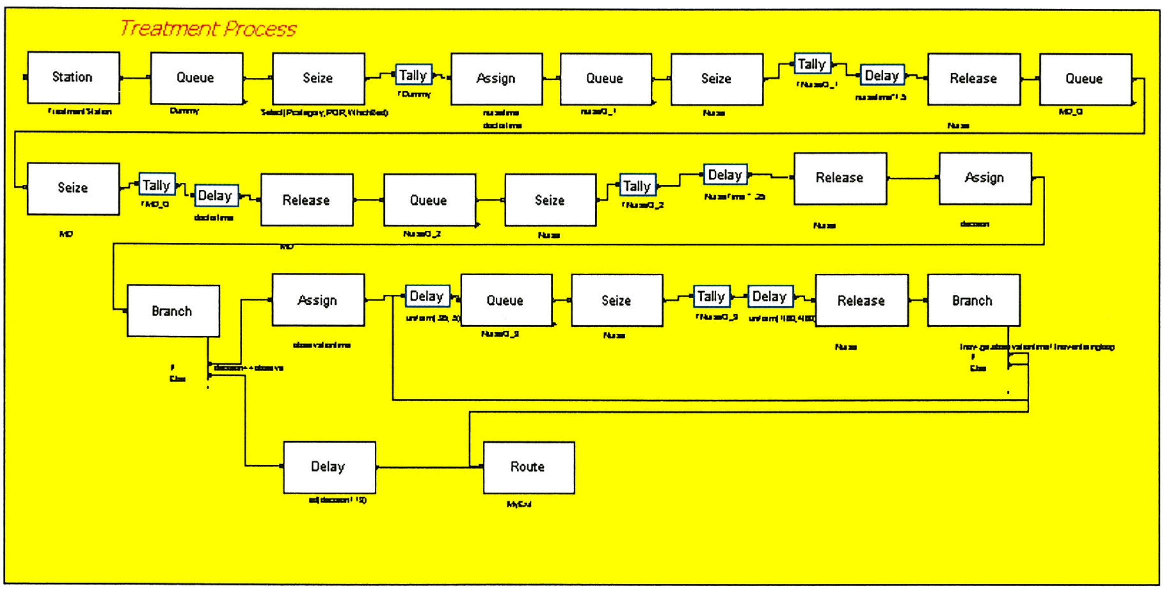


Figure 19: Treatment Station

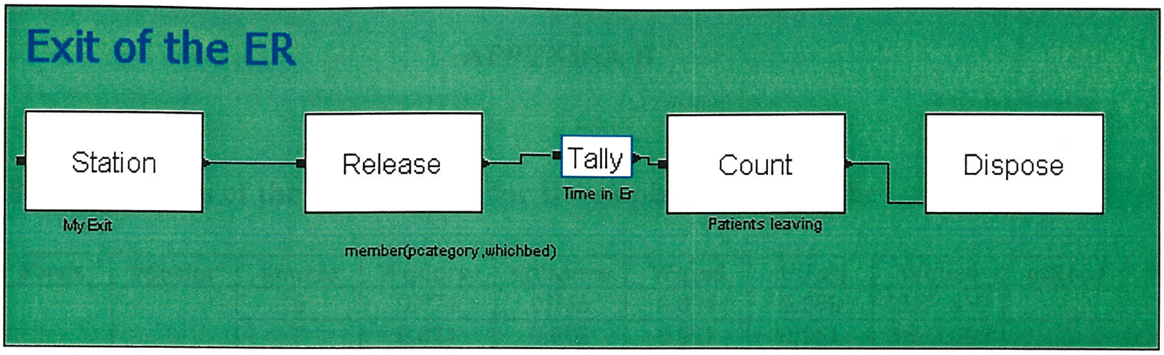


Figure 20: Exit of the System

APPENDIX B

B.1 Outputs of the Experiments for Developing the Heuristic

#beds	#doctor	#nurse	Wqnurse	utilnurse	Wqmd	utilmd	Wqbed	utilbed
10	1	6	0.074	0.765	0.094	0.588	2887.850	1
		7	0.024	0.673	0.091	0.561	3603.660	1
		8	0.004	0.581	0.110	0.601	2948.740	1
		9	0.001	0.525	0.099	0.577	2521.590	1
		10	0.000	0.468	0.108	0.597	3177.710	1
	2	6	0.097	0.793	0.006	0.284	3006.150	1
		7	0.028	0.693	0.007	0.293	3392.950	1
		8	0.006	0.600	0.009	0.312	3047.180	1
		9	0.001	0.544	0.008	0.298	2350.090	1
		10	0.000	0.487	0.008	0.295	2824.690	1
	3	6	0.110	0.811	0.000	0.171	3195.050	1
		7	0.026	0.684	0.001	0.204	2685.480	1
		8	0.005	0.598	0.001	0.215	2443.980	1
		9	0.001	0.544	0.001	0.203	2376.600	1
		10	0.000	0.481	0.001	0.211	2205.980	1
	4	6	0.091	0.788	0.000	0.150	2464.820	1
		7	0.028	0.695	0.000	0.148	3304.820	1
		8	0.007	0.605	0.000	0.152	2612.600	1
		9	0.001	0.538	0.000	0.157	2833.890	1
		10	0.000	0.490	0.000	0.153	2378.620	1

#beds	#doctor	#nurse	Wqnurse	utilnurse	Wqmd	utilmd	Wqbed	utilbed
20	1	6	0.877	0.952	0.604	0.741	1975.290	1
		7	0.499	0.877	0.684	0.773	1450.450	1
		8	0.303	0.813	0.683	0.797	1216.260	1
		9	0.158	0.749	0.694	0.820	1010.740	1
		10	0.081	0.693	0.683	0.833	802.210	1
	2	6	1.382	0.998	0.018	0.401	1528.440	1
		7	0.804	0.987	0.017	0.427	562.930	1
		8	0.396	0.958	0.024	0.465	10.429	0.987
		9	0.155	0.879	0.025	0.480	2.358	0.940
		10	0.066	0.800	0.028	0.486	1.642	0.920
	3	6	1.483	0.998	0.003	0.250	1421.060	1
		7	0.831	0.989	0.003	0.285	578.530	1
		8	0.421	0.959	0.004	0.312	10.892	0.985
		9	0.156	0.881	0.003	0.324	2.132	0.936
		10	0.063	0.805	0.004	0.326	1.809	0.923
	4	6	1.412	0.998	0.000	0.193	1597.880	1
		7	0.831	0.989	0.000	0.215	645.560	1
		8	0.425	0.961	0.001	0.234	13.616	0.990
		9	0.159	0.881	0.000	0.242	1.978	0.932
		10	0.065	0.808	0.000	0.246	1.996	0.922

#beds	#doctor	#nurse	Wqnurse	utilnurse	Wqmd	utilmd	Wqbed	utilbed
30	1	3	10.327	1	0.050	0.483	2960.650	1
		4	6.686	1	0.079	0.544	3079.150	1
		5	4.488	1	0.168	0.645	2110.460	1
		6	1.728	0.958	1.474	0.720	1917.910	1
		7	1.176	0.883	1.633	0.770	1383.640	1
		8	0.877	0.818	1.645	0.803	1035.370	1
		9	0.638	0.773	1.601	0.847	506.160	0.999
		10	0.329	0.751	1.452	0.915	21.868	0.975
		11	0.287	0.676	1.632	0.906	142.267	0.999
		12	0.154	0.646	1.521	0.932	44.585	0.980
		13	0.032	0.612	1.320	0.960	3.684	0.928
	2	3	10.359	1	0.003	0.236	3048.700	1
		4	6.838	1	0.010	0.295	3222.630	1
		5	4.782	1	0.014	0.344	2354.930	1
		6	3.463	1	0.020	0.405	1359.330	1
		7	2.450	1	0.026	0.444	339.238	1.000
		8	1.281	0.977	0.037	0.497	2.305	0.889
		9	0.517	0.912	0.031	0.513	0.985	0.763
		10	0.240	0.843	0.032	0.521	0.772	0.719
		11	0.104	0.772	0.038	0.527	0.568	0.692
		12	0.042	0.705	0.039	0.522	0.481	0.675
		13	0.018	0.656	0.042	0.525	0.595	0.678
	3	3	10.190	1	0.000	0.141	5031.000	1.000
		4	6.846	1	0.001	0.188	3462.030	1
		5	4.776	1	0.002	0.220	2308.750	1
		6	3.450	1	0.004	0.266	1390.570	1
		7	2.504	1	0.006	0.298	371.727	1
		8	1.311	0.976	0.007	0.331	1.946	0.882
		9	0.557	0.913	0.005	0.344	0.970	0.763
		10	0.230	0.836	0.004	0.348	0.629	0.701
		11	0.108	0.773	0.005	0.349	0.576	0.686
		12	0.052	0.715	0.005	0.351	0.552	0.680
		13	0.021	0.655	0.006	0.348	0.483	0.664
	4	3	10.190	1	0.000	0.106	5031	1
		4	6.832	1	0.000	0.145	2601.060	1
		5	4.821	1	0.000	0.180	2040.300	1
		6	3.465	1	0.001	0.193	1453.390	1
		7	2.510	1	0.001	0.220	325.595	1.000
		8	1.342	0.979	0.002	0.248	1.937	0.891
		9	0.552	0.916	0.001	0.258	1.037	0.766
		10	0.239	0.838	0.001	0.259	0.657	0.707
		11	0.113	0.776	0.001	0.263	0.615	0.689
		12	0.045	0.707	0.001	0.262	0.465	0.670
		13	0.020	0.656	0.001	0.263	0.494	0.669

#beds	#doctor	#nurse	Wqnurse	utilnurse	Wqmd	utilmd	Wqbed	utilbed
40	1	6	2.506	0.957	2.498	0.726	1523.41	1
		7	1.830	0.884	2.637	0.772	1182.34	1
		8	1.302	0.963	1.488	0.966	6.260	0.936
		9	0.477	0.878	1.917	0.977	2.048	0.891
		10	0.239	0.799	2.089	0.979	3.623	0.891
	2	7	4.055	1	0.030	0.449	263.553	1
		8	2.184	0.985	0.047	0.508	2.028	0.839
		9	0.789	0.915	0.037	0.520	0.995	0.634
		10	0.310	0.842	0.035	0.526	0.992	0.558
	3	6	5.385	1	0.005	0.257	1390.390	1
		7	4.109	1	0.007	0.301	262.674	1.000
		8	2.161	0.983	0.010	0.337	2.021	0.827
		9	0.842	0.924	0.007	0.351	1.059	0.646
		10	0.312	0.843	0.005	0.349	0.659	0.553
	4	6	5.417	1.00	0.001	0.195	1274.610	1.000
		7	4.103	1.00	0.002	0.226	332.693	1.000
		8	2.137	0.983	0.002	0.252	2.026	0.823
		9	0.829	0.924	0.001	0.265	1.096	0.638
		10	0.350	0.850	0.001	0.265	0.698	0.564

APPENDIX C

C.1 Surface Graphs for: beds = 10

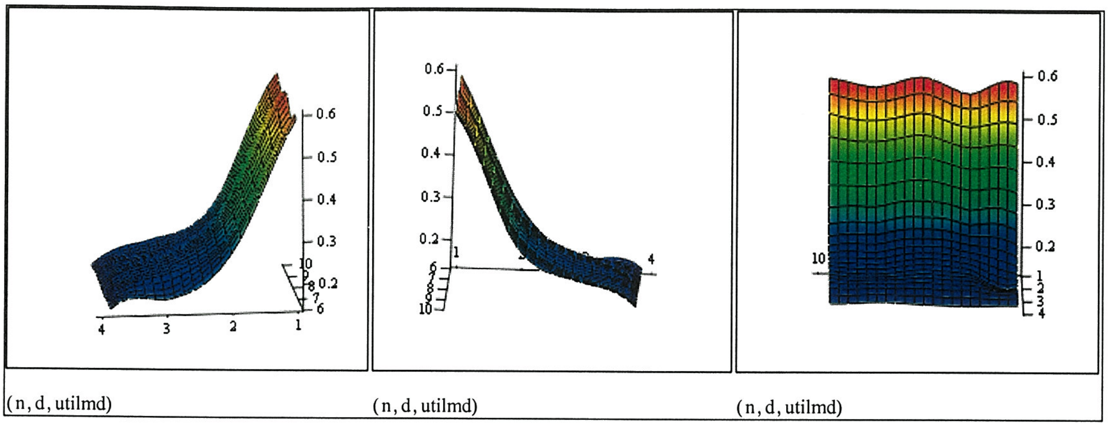


Figure 21: RS of nurses, doctors and utilization of doctor

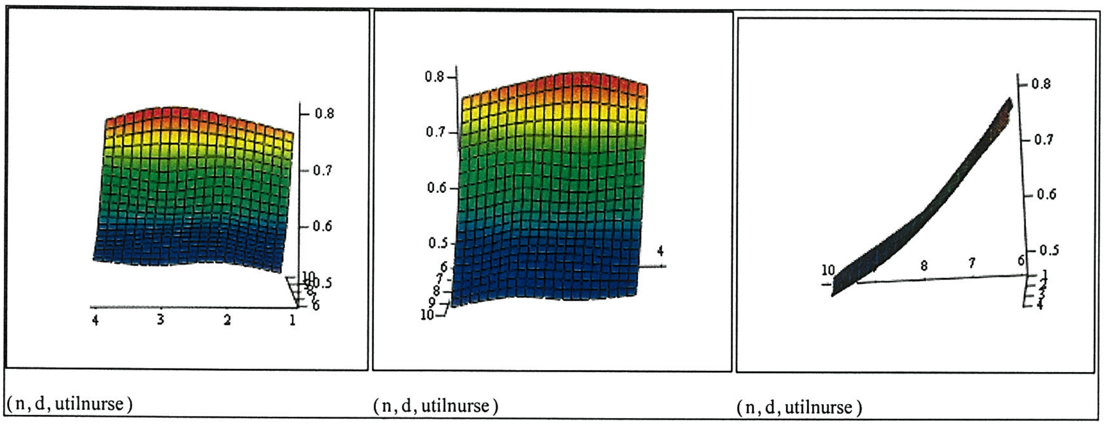


Figure 22: RS of nurses, doctors and utilization of nurse

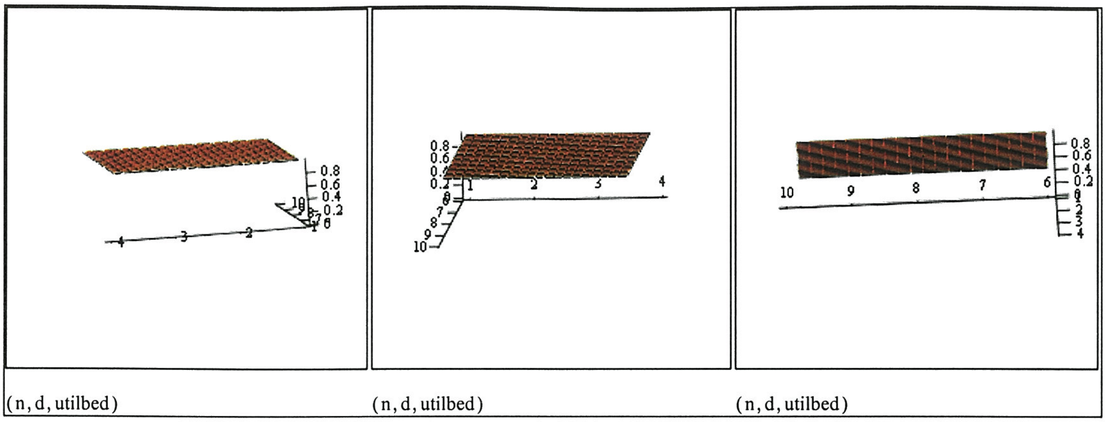


Figure 23: RS of nurses, doctors and utilization of beds

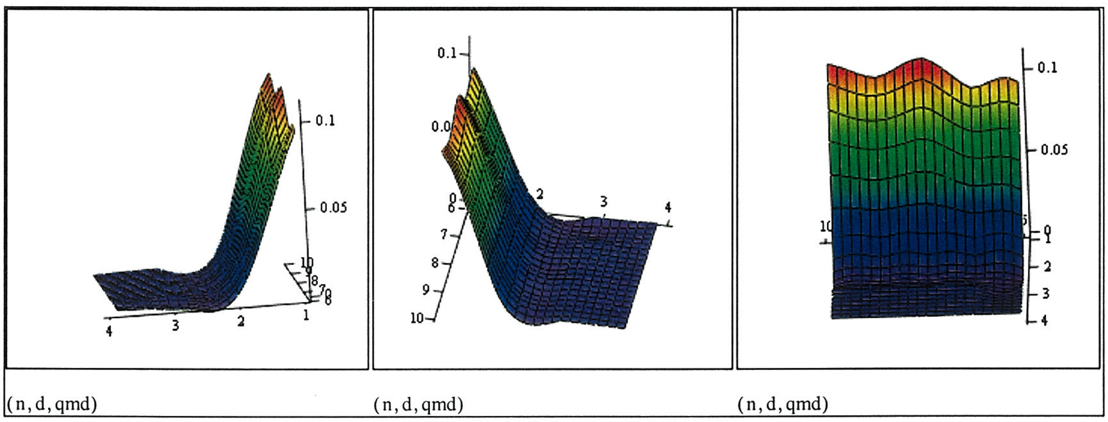


Figure 24: RS of nurses, doctors and time in doctor queue

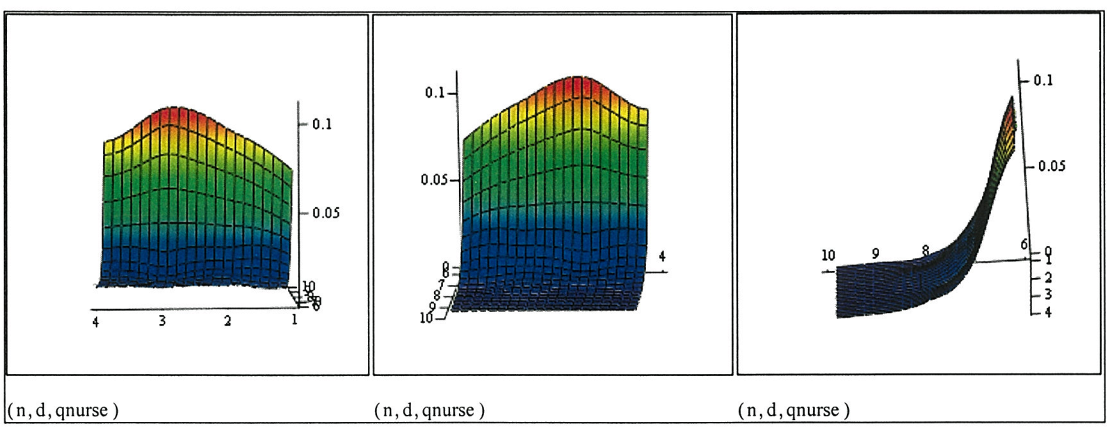


Figure 25: RS of nurses, doctors and time in nurse queue

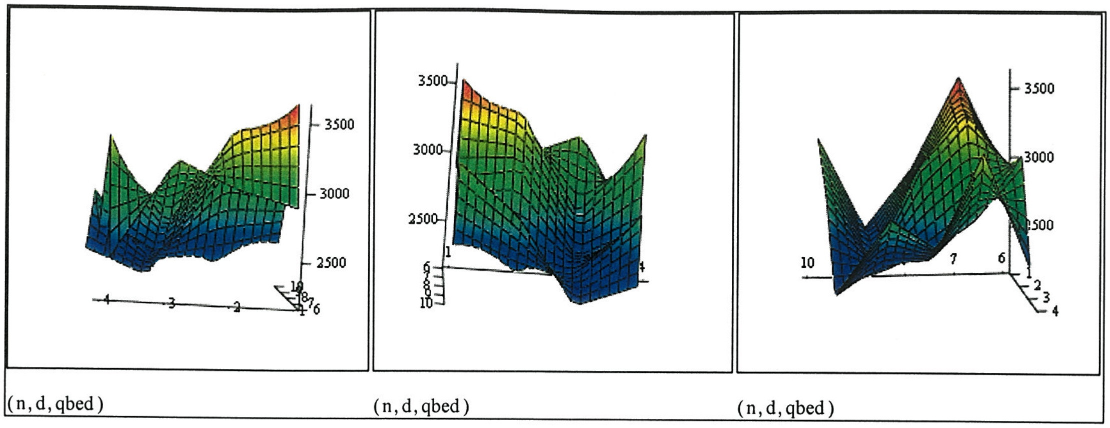


Figure 26: RS of nurses, doctors and time in bed queue

C.2 Surface Graphs for: beds = 20

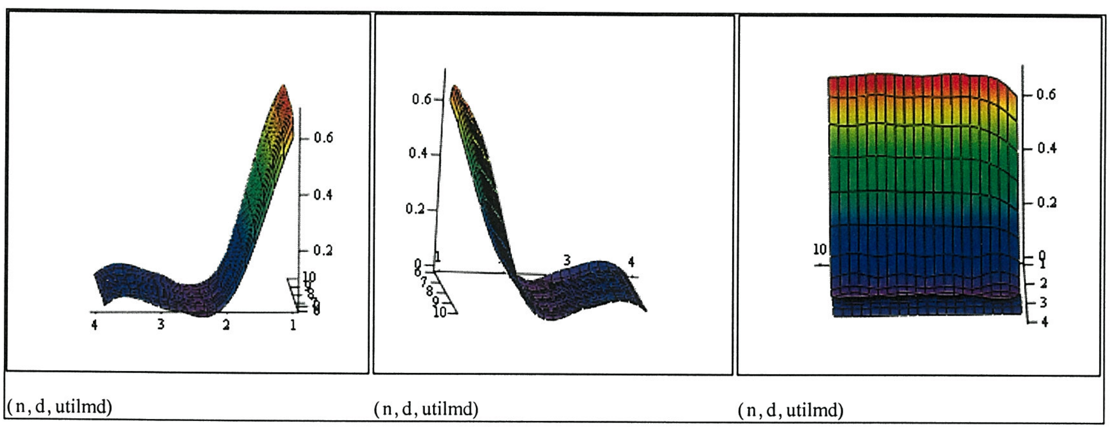


Figure 27: RS of nurses, doctors and utilization of doctor

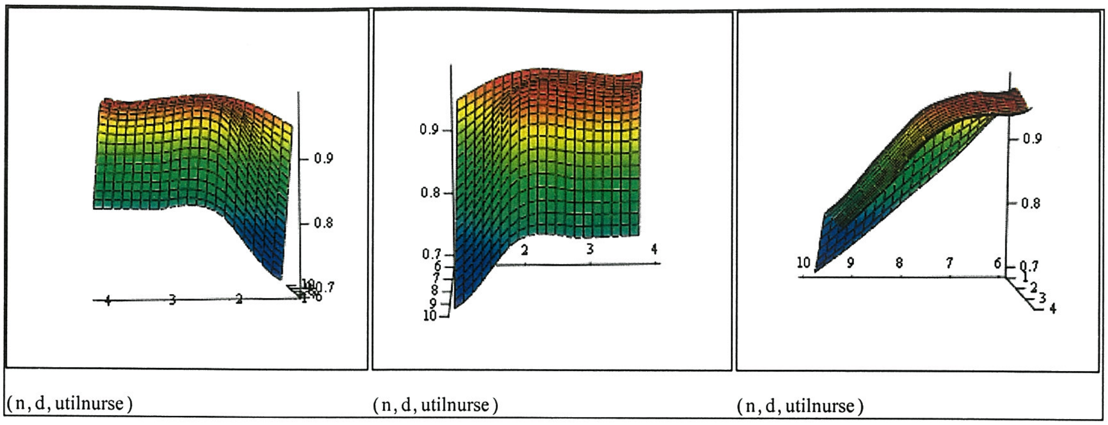


Figure 28: RS of nurses, doctors and utilization of nurse

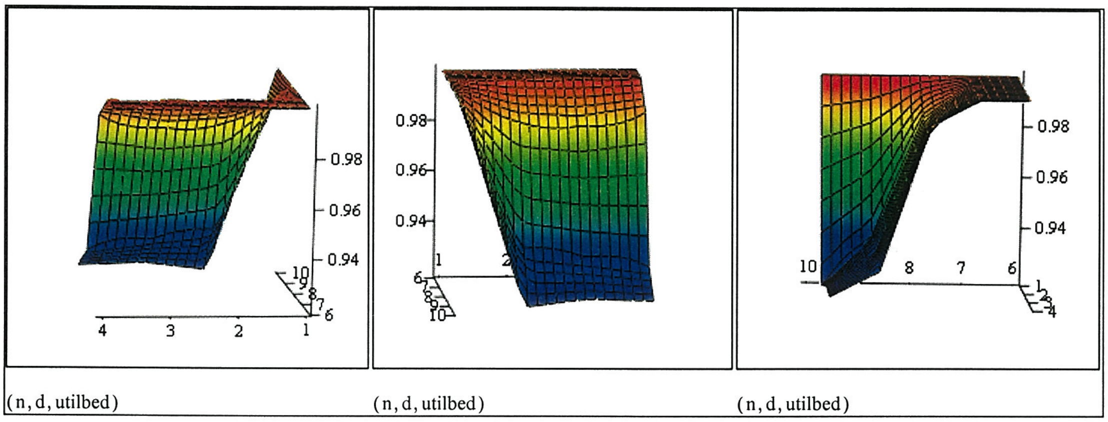


Figure 29: RS of nurses, doctors and utilization of beds

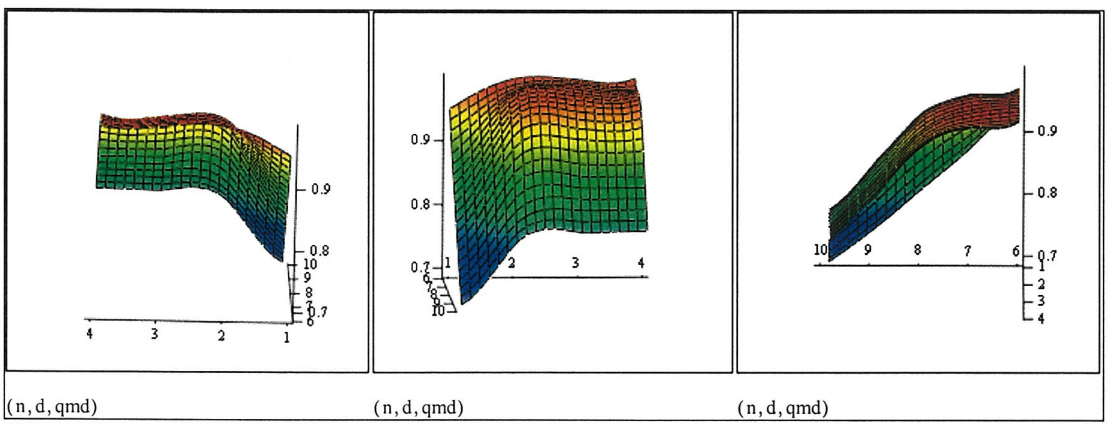


Figure 30: RS of nurses, doctors and time in doctor queue

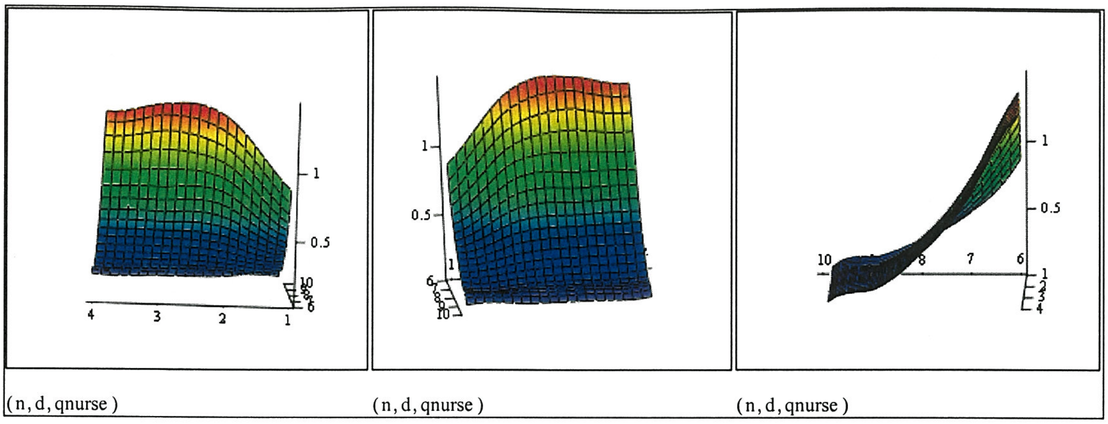


Figure 31: RS of nurses, doctors and time in nurse queue

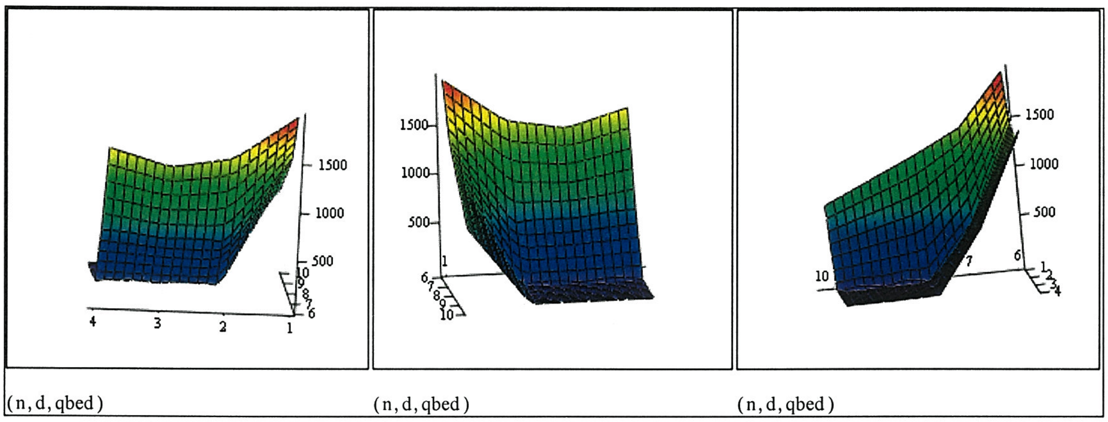
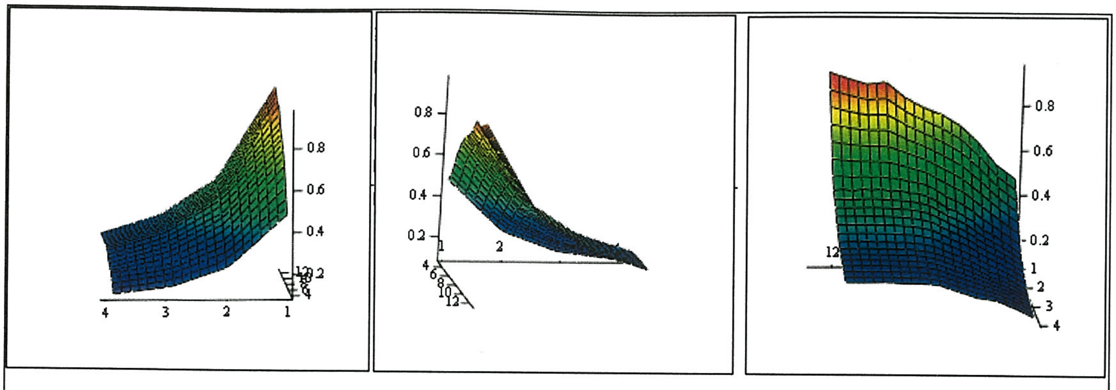


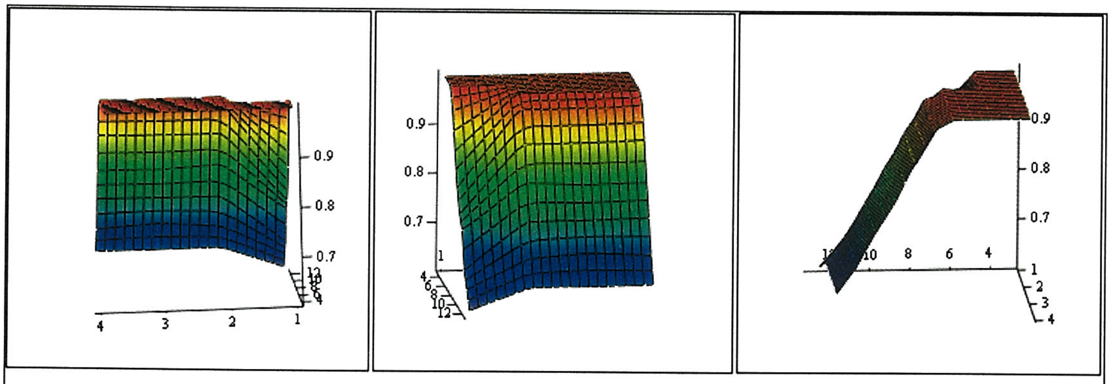
Figure 32: RS of nurses, doctors and time in bed queue

C.3 Surface Graphs for: beds = 30



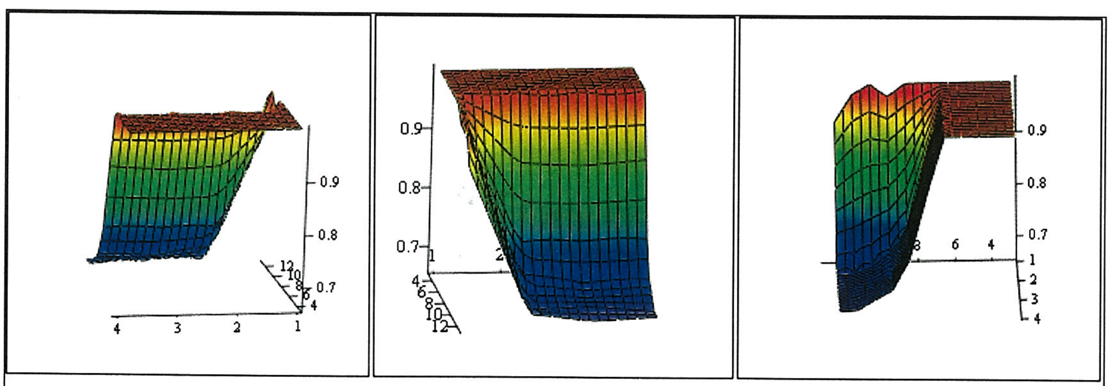
(n, d, utilmd) (n, d, utilmd) (n, d, utildoc)

Figure 33: RS of nurses, doctors and utilization of doctors



(n, d, utilnurse) (n, d, utilnurse) (n, d, utilnurse)

Figure 34: RS of nurses, doctors and utilization of nurse



(n, d, utilbed) (n, d, utilbed) (n, d, utilbed)

Figure 35: RS of nurses, doctors and utilization of beds

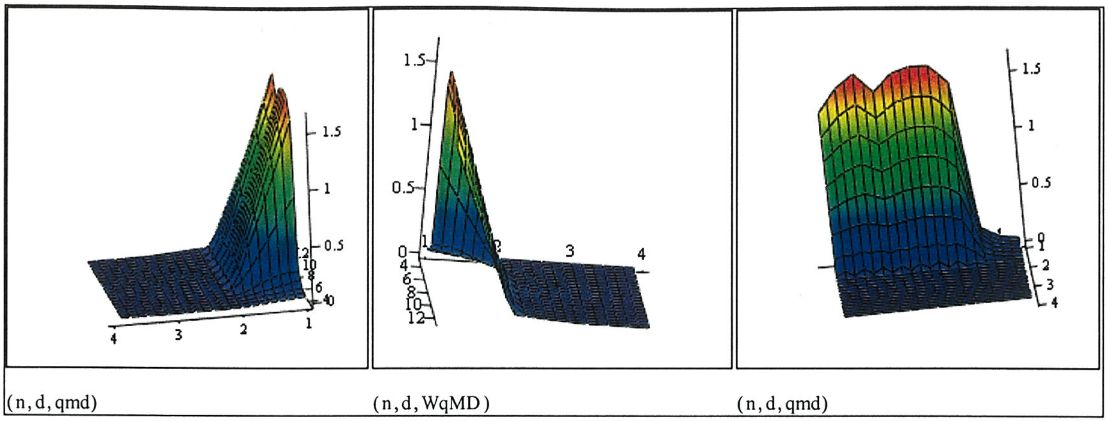


Figure 36: RS of nurses, doctors and time in doctor queue

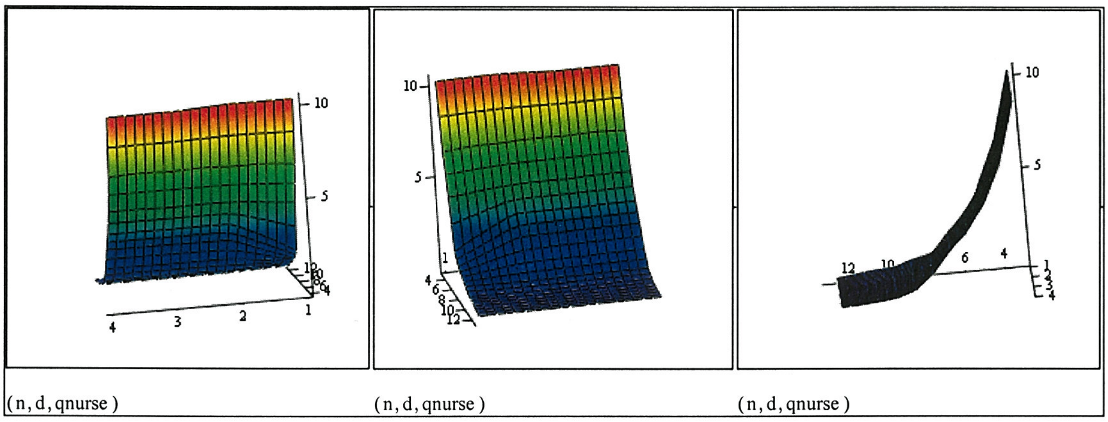


Figure 37: RS of nurses, doctors and time in nurse queue

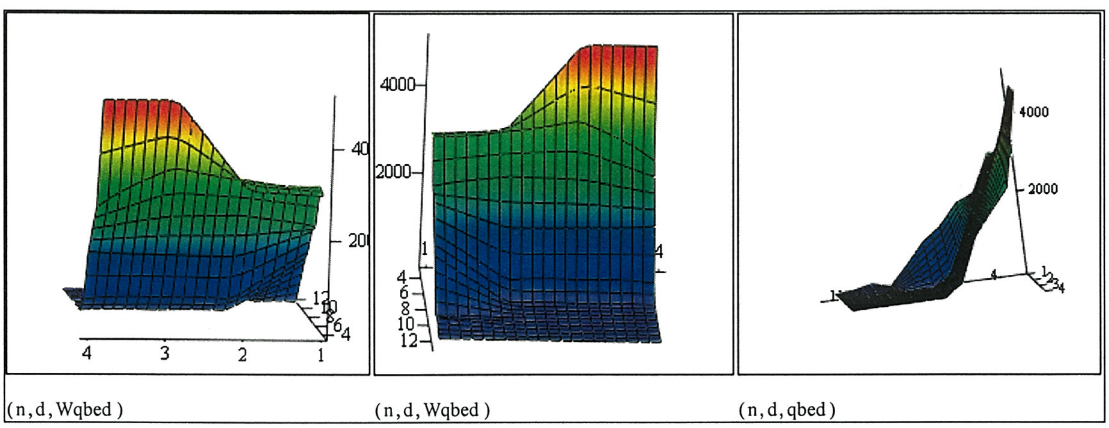


Figure 38: RS of nurses, doctors and time in bed queue

C.4 Surface Graphs for: beds = 40

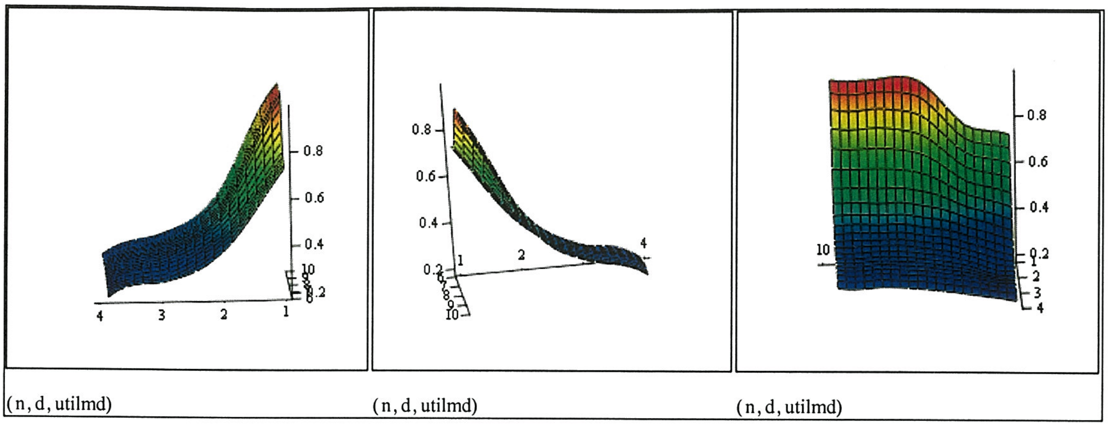


Figure 39: RS of nurses, doctors and utilization of doctor

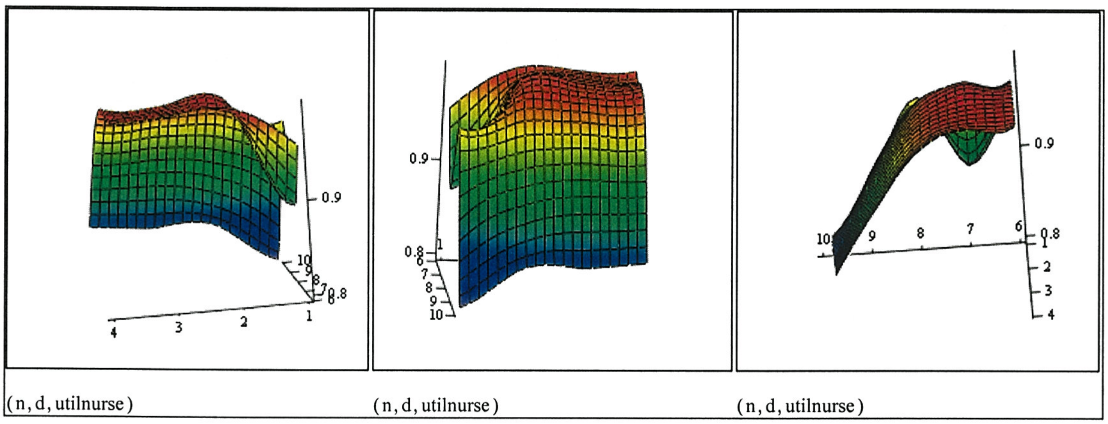


Figure 40: RS of nurses, doctors and utilization of nurse

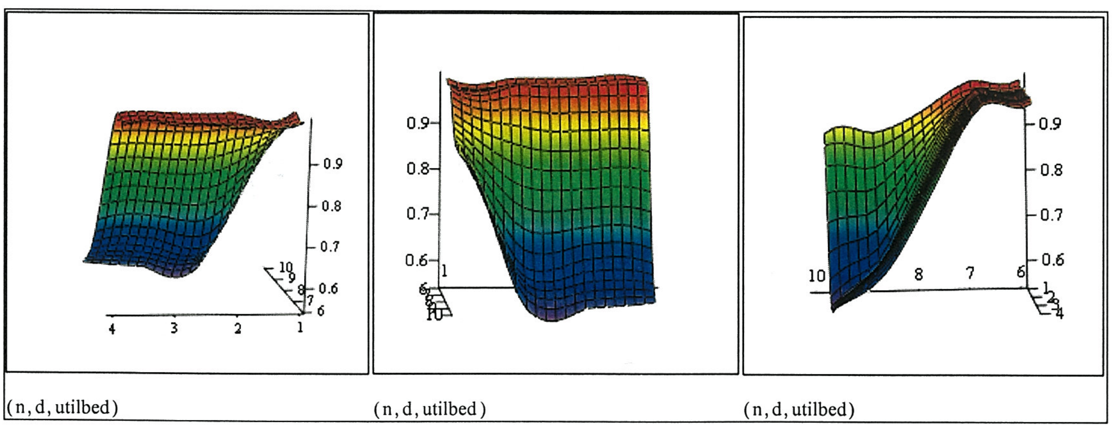


Figure 41: RS of nurses, doctors and utilization of beds

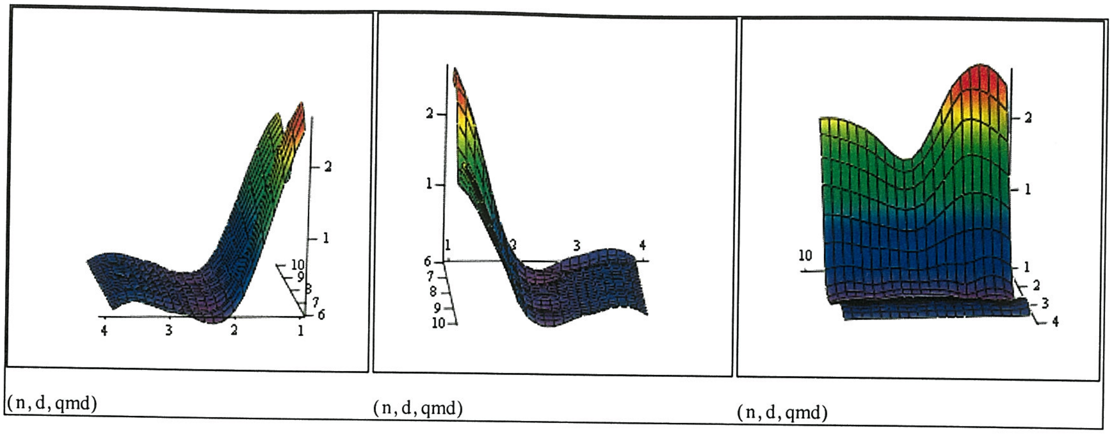


Figure 42: RS of nurses, doctors and time in doctor queue

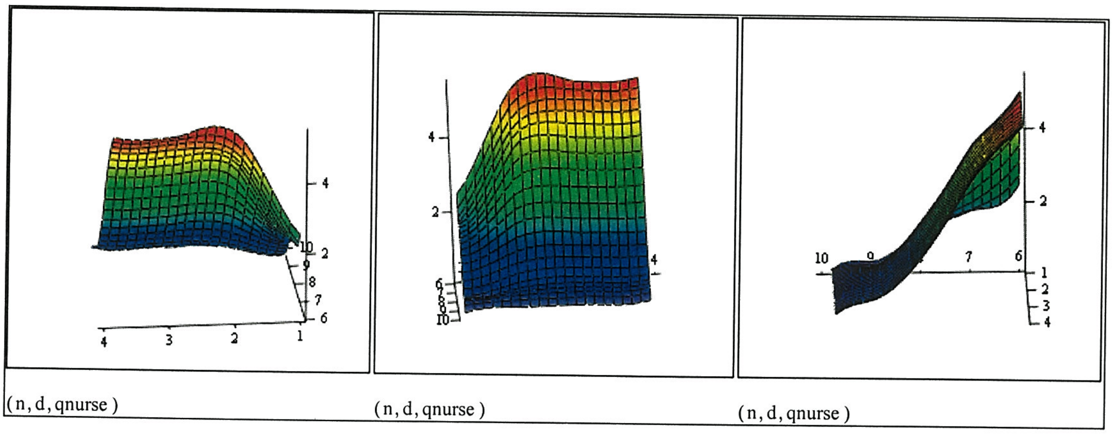


Figure 43: RS of nurses, doctors and time in nurse queue

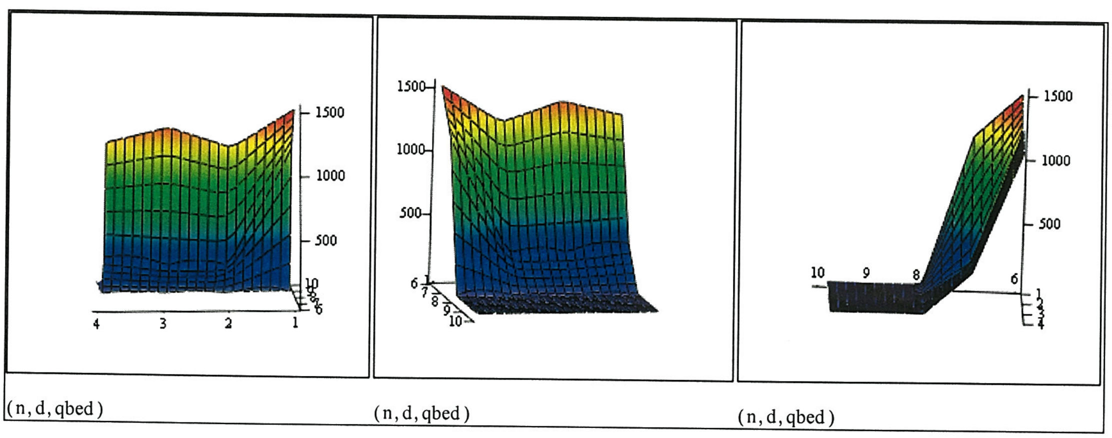


Figure 44: RS of nurses, doctors and time in bed queue

APPENDIX D

D.1 2-D Regression Graphs for ρ_{MD}

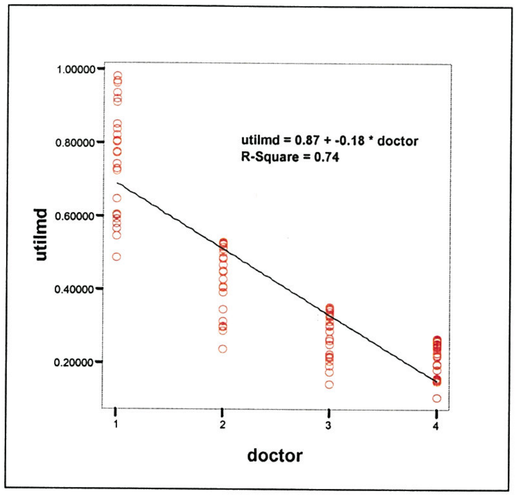


Figure 45: Regression Line for ρ_{MD} versus doctor

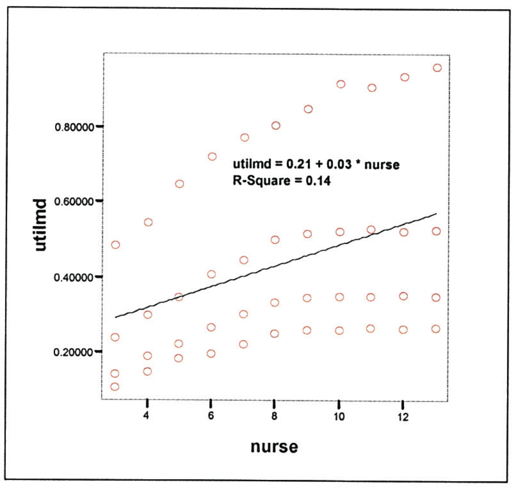


Figure 46: Regression line for ρ_{MD} versus nurse

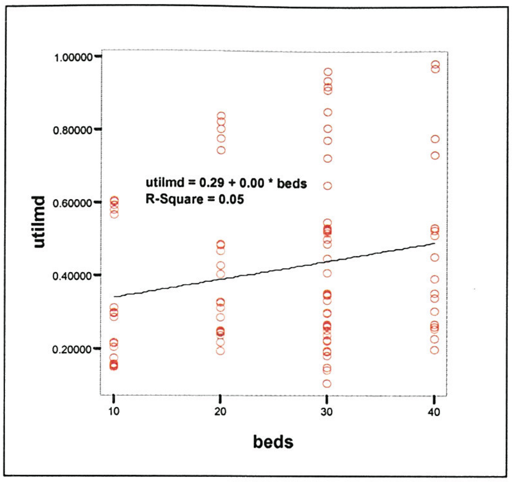


Figure 47: Regression line for ρ_{MD} versus beds

D.2 2-D Regression Graphs for ρ_{RN}

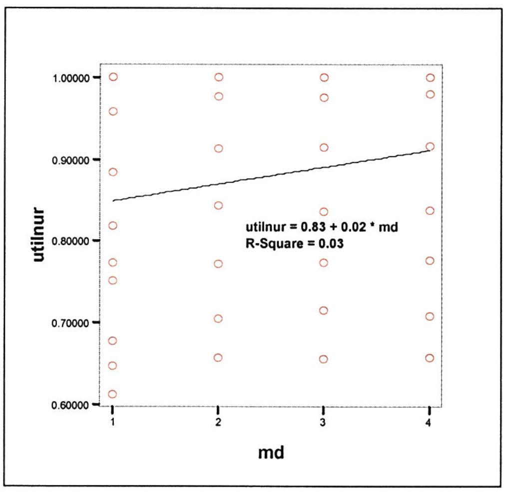


Figure 48: Regression line for ρ_{RN} versus doctor

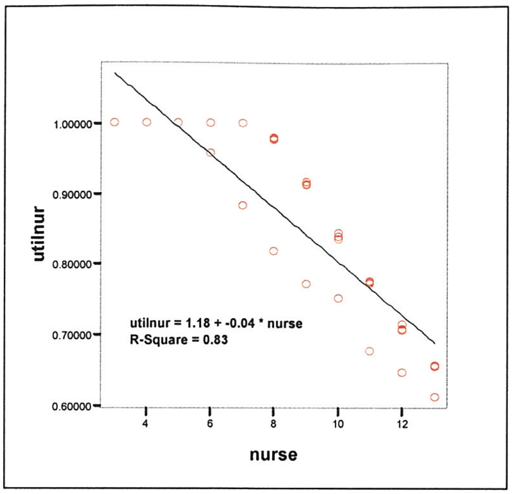


Figure 49: Regression line for ρ_{RN} versus nurse

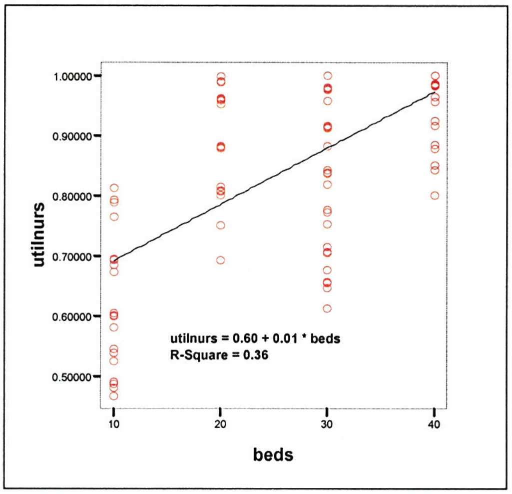


Figure 50: Regression line for ρ_{RN} versus beds

D.3 2-D Regression Graphs for ρ_{beds}

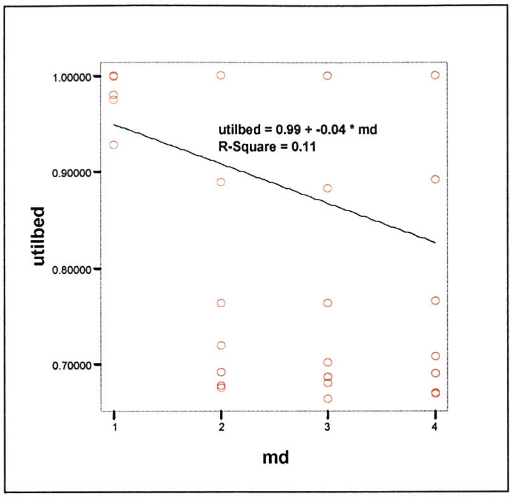


Figure 51: Regression line for ρ_{beds} versus doctor

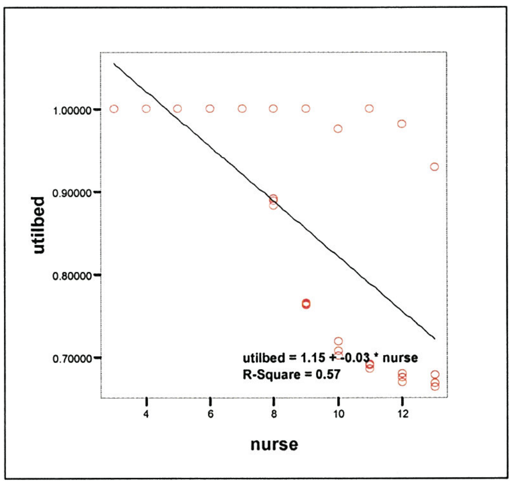


Figure 52: Regression line for ρ_{beds} versus nurse

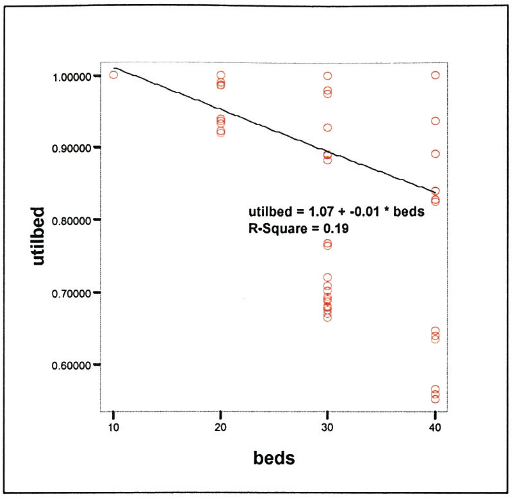


Figure 53: Regression line for ρ_{beds} versus beds

D.4 2-D Regression Graphs for W_{qMD}

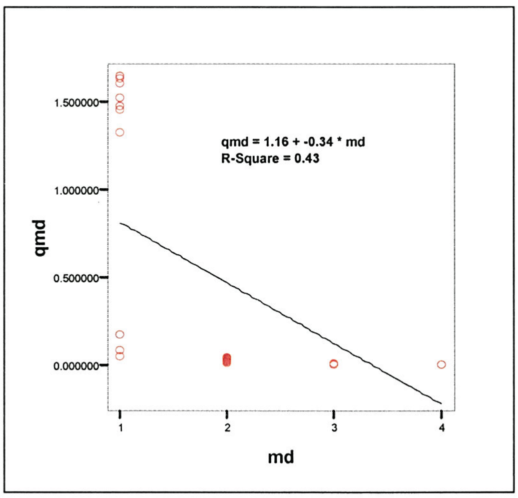


Figure 54: Regression line for W_{qMD} versus doctor

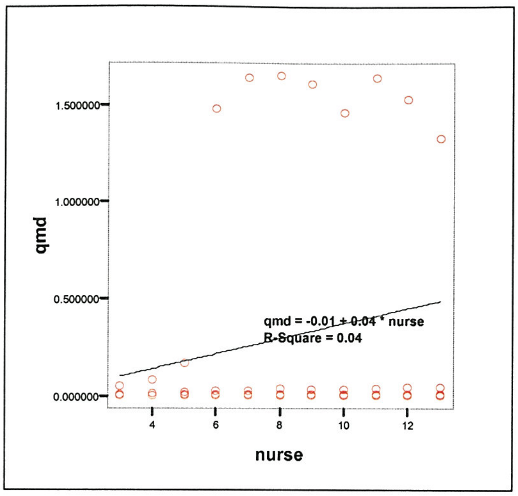


Figure 55: Regression line for W_{qMD} versus nurse

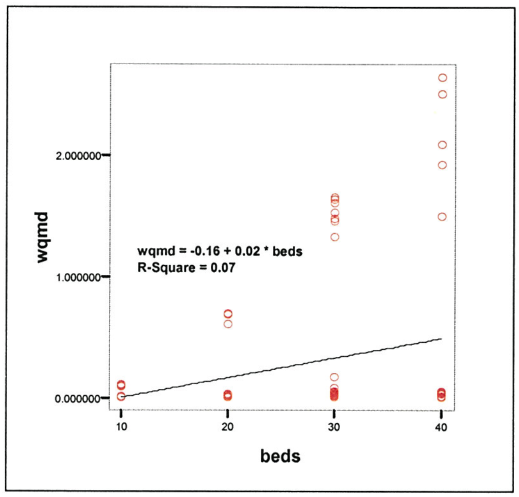


Figure 56: Regression line for W_{qMD} versus beds

D.5 2-D Regression Graphs for W_{qRN}

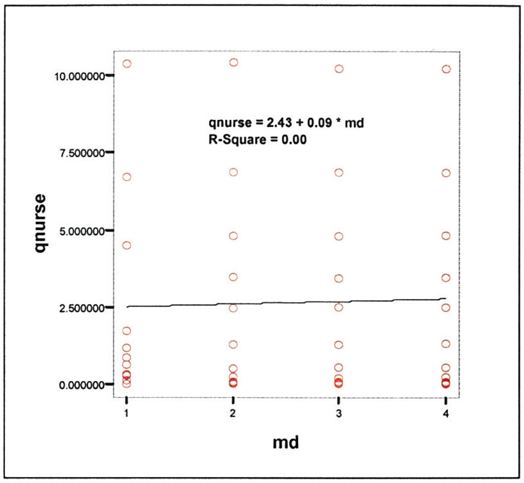


Figure 57: Regression line for W_{qRN} versus doctor

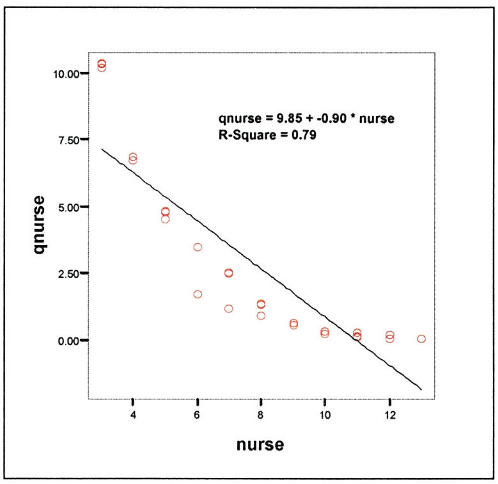


Figure 58: Regression line for W_{qRN} versus nurse

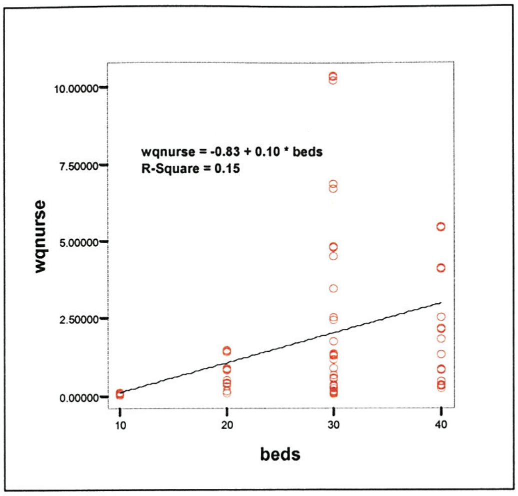


Figure 59: Regression line for W_{qRN} versus beds

D.6 2-D Regression Graphs for W_{qbeds}

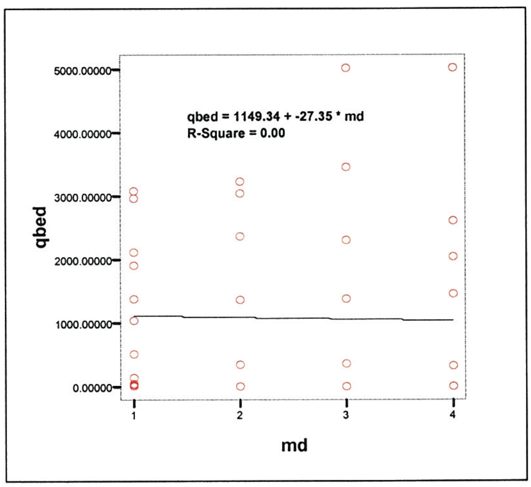


Figure 60: Regression line for W_{qbeds} versus doctor

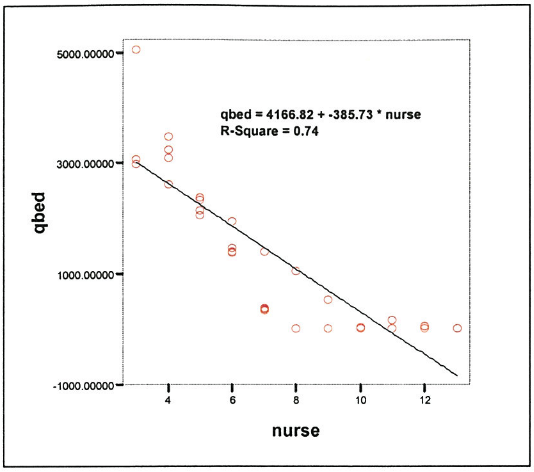


Figure 61: Regression line for W_{qbeds} versus nurse

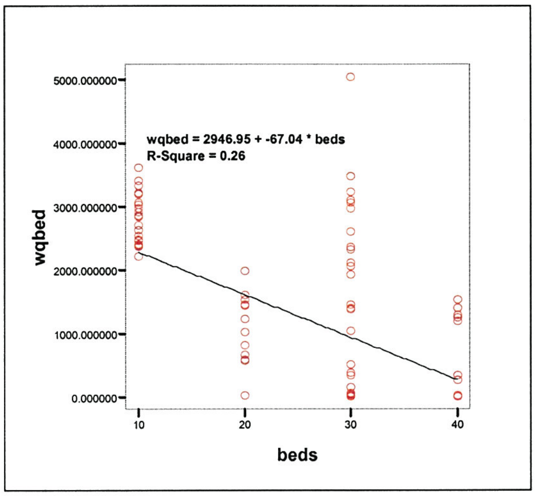


Figure 62: Regression line for W_{qbeds} versus beds

APPENDIX E

Table 31: Possible Scenarios for W_{qRN} and ρ_{RN}

W_{qRN}	ρ_{RN}	Decision	Rule #
1.	1.	ADD 2 MD	1
	2.	ADD 1 MD	2
	3.	Goals Met	3
	4.	Goals Met	4
	5.	Goals Met	5
	6.	Goals Met	6
2.	1.	ADD 1 MD	7
	2.	ADD 1 MD	8
	3.	Goals Met	9
	4.	Goals Met	10
	5.	Goals Met	11
	6.	Goals Met	12
3.	1.	ADD 1 MD	13
	2.	ADD 1 MD	14
	3.	Goals Met	15
	4.	Goals Met	16
	5.	Goals Met	17
	6.	Goals Met	18
4.	1.	ADD 1 MD	19
	2.	ADD 1 MD	20
	3.	Goals Met	21
	4.	Goals Met	22
	5.	Goals Met	23
	6.	Goals Met	24
5.	1.	OVH	25
	2.	ADD 1 MD + 1 RN	26
	3.	ADD 1 MD + 1 RN	27
	4.	ADD 1 MD + 1 RN	28
	5.	ADD 1 MD + 1 RN	29
	6.	ADD 1 MD + 1 RN + 1 BED	30
6.	1.	OVH	31
	2.	ADD 1 MD + 1 RN	32
	3.	ADD 1 MD + 1 RN	33
	4.	ADD 1 MD + 1 RN	34
	5.	ADD 1 MD + 1 RN	35
	6.	ADD 1 MD + 1 RN + 1 BED	36

Table 32: Possible Scenarios for W_{qRN} and ρ_{MD}

W_{qRN}	ρ_{MD}	Decision	Rule #
1.	1.	ADD 2 BEDS	37
	2.	ADD 1 BED	38
	3.	Goals Met	39
	4.	Goals Met	40
	5.	Goals Met	41
	6.	Goals Met	42
2.	1.	ADD 1 BED	43
	2.	ADD 1 BED	44
	3.	Goals Met	45
	4.	Goals Met	46
	5.	Goals Met	47
	6.	Goals Met	48
3.	1.	ADD 1 RN + 3 BEDS	49
	2.	ADD 1 RN + 2 BEDS	50
	3.	Goals Met	51
	4.	Goals Met	52
	5.	Goals Met	53
	6.	Goals Met	54
4.	1.	ADD 1 RN + 3 BEDS	55
	2.	ADD 1 RN + 2 BEDS	56
	3.	Goals Met	57
	4.	Goals Met	58
	5.	Goals Met	59
	6.	Goals Met	60
5.	1.	ADD 1 RN + 3 BEDS	61
	2.	ADD 1 RN	62
	3.	ADD 1 RN	63
	4.	ADD 1 RN	64
	5.	ADD 1 RN	65
	6.	ADD 1 RN	66
6.	1.	ADD 2 RN + 3 BEDS	67
	2.	ADD 2 RN	68
	3.	ADD 2 RN	69
	4.	ADD 2 RN	70
	5.	ADD 2 RN	71
	6.	ADD 2 RN	72

Table 33: Possible Scenarios for W_{qRN} and ρ_{bed}

W_{qRN}	ρ_{bed}	Decision	Rule #
1.	1.	ADD 4 RN	73
	2.	ADD 2 RN	74
	3.	Goals Met	75
	4.	Goals Met	76
	5.	Goals Met	77
	6.	Goals Met	78
2.	1.	ADD 4 RN	79
	2.	ADD 2 RN	80
	3.	Goals Met	81
	4.	Goals Met	82
	5.	Goals Met	83
	6.	Goals Met	84
3.	1.	ADD 4 RN	85
	2.	ADD 2 RN	86
	3.	Goals Met	87
	4.	Goals Met	88
	5.	Goals Met	89
	6.	Goals Met	90
4.	1.	ADD 4 RN	91
	2.	ADD 2 RN	92
	3.	Goals Met	93
	4.	Goals Met	94
	5.	Goals Met	95
	6.	Goals Met	96
5.	1.	ADD 4 RN	97
	2.	ADD 2 RN	98
	3.	ADD 1 RN + 1 MD	99
	4.	ADD 1 RN + 1 MD	100
	5.	ADD 1 RN + 1 MD	101
	6.	ADD 1 RN + 1 MD	102
6.	1.	ADD 4 RN	103
	2.	ADD 4 RN + 1 MD	104
	3.	ADD 4 RN + 1 MD	105
	4.	ADD 4 RN + 1 MD	106
	5.	ADD 4 RN + 1 MD	107
	6.	ADD 4 RN + 1 MD	108

Table 34: Possible Scenarios for W_{qRN} and W_{qMD}

W_{qRN}	W_{qMD}	Decision	Rule #
1.	1.	Goals Met	109
	2.	Goals Met	110
	3.	Goals Met	111
	4.	Goals Met	112
	5.	ADD 1 MD	113
	6.	ADD 1 MD	114
2.	1.	Goals Met	115
	2.	Goals Met	116
	3.	Goals Met	117
	4.	Goals Met	118
	5.	ADD 1 MD + 1 RN	119
	6.	ADD 1 MD + 1 RN	120
3.	1.	Goals Met	121
	2.	Goals Met	122
	3.	Goals Met	123
	4.	Goals Met	124
	5.	ADD 1 MD + 1 RN	125
	6.	ADD 1 MD + 1 RN	126
4.	1.	Goals Met	127
	2.	Goals Met	128
	3.	Goals Met	129
	4.	Goals Met	130
	5.	ADD 1 MD + 1 RN	131
	6.	ADD 1 MD + 1 RN	132
5.	1.	ADD 1 RN	133
	2.	ADD 1 MD + 1 RN	134
	3.	ADD 1 MD + 1 RN	135
	4.	ADD 1 MD + 1 RN	136
	5.	ADD 1 MD + 1 RN	137
	6.	ADD 1 MD + 1 RN	138
6.	1.	ADD 1 MD + 3 RN	139
	2.	ADD 1 MD + 2 RN	140
	3.	ADD 1 MD + 2 RN	141
	4.	ADD 1 MD + 2 RN	142
	5.	ADD 1 MD + 2 RN	143
	6.	ADD 1 MD + 3 RN	144

Table 35: Possible Scenarios for W_{qRN} and W_{qbeds}

W_{qRN}	W_{qbeds}	Decision	Rule #
1.	1.	Goals Met	145
	2.	Goals Met	146
	3.	Goals Met	147
	4.	Goals Met	148
	5.	ADD 2 RN	149
	6.	ADD 2 RN + 1 BED	150
2.	1.	Goals Met	151
	2.	Goals Met	152
	3.	Goals Met	153
	4.	Goals Met	154
	5.	ADD 1 RN + 1 BED	155
	6.	ADD 1 MD + 2 RN	156
3.	1.	Goals Met	157
	2.	Goals Met	158
	3.	Goals Met	159
	4.	Goals Met	160
	5.	ADD 1 MD + 2 RN	161
	6.	ADD 2 MD + 3 RN	162
4.	1.	Goals Met	163
	2.	Goals Met	164
	3.	Goals Met	165
	4.	Goals Met	166
	5.	ADD 1 RN + 2 BEDS	167
	6.	ADD 2 RN + 2 BEDS	168
5.	1.	ADD 1 RN	169
	2.	ADD 1 RN	170
	3.	ADD 1 RN	171
	4.	ADD 1 RN	172
	5.	ADD 1 RN + 1 BED	173
	6.	ADD 1 MD + 2 RN + 1 BED	174
6.	1.	ADD 2 RN	175
	2.	ADD 2 RN	176
	3.	ADD 2 RN	177
	4.	ADD 2 RN	178
	5.	ADD 1 MD + 1 RN	179
	6.	ADD 2 RN	180

Table 36: Possible Scenarios for W_{qMD} and ρ_{RN}

W_{qMD}	ρ_{RN}	Decision	Rule #
1.	1.	ADD 1 MD + 10 BEDS	181
	2.	ADD 1 MD + 7 BEDS	182
	3.	Goals Met	183
	4.	Goals Met	184
	5.	Goals Met	185
	6.	Goals Met	186
2.	1.	ADD 1 MD + 10 BEDS	187
	2.	ADD 1 MD + 7 BEDS	188
	3.	Goals Met	189
	4.	Goals Met	190
	5.	Goals Met	191
	6.	Goals Met	192
3.	1.	ADD 1 MD + 10 BEDS	193
	2.	ADD 1 MD + 7 BEDS	194
	3.	Goals Met	195
	4.	Goals Met	196
	5.	Goals Met	197
	6.	Goals Met	198
4.	1.	ADD 1 MD + 10 BEDS	199
	2.	ADD 1 MD + 7 BEDS	200
	3.	Goals Met	201
	4.	Goals Met	202
	5.	Goals Met	203
	6.	Goals Met	204
5.	1.	ADD 1 MD + 10 BEDS	205
	2.	ADD 1 MD + 7 BEDS	206
	3.	ADD 1 MD + 1 RN	207
	4.	ADD 1 MD + 1 RN	208
	5.	ADD 1 MD + 1 RN	209
	6.	ADD 1 MD + 2 RN	210
6.	1.	ADD 1 MD + 10 BEDS	211
	2.	ADD 1 MD + 8 BEDS	212
	3.	ADD 1 MD + 1 BED	213
	4.	ADD 1 MD + 1 BED	214
	5.	ADD 1 MD + 1 BED	215
	6.	ADD 1 MD + 1 BED	216

Table 37: Possible Scenarios for W_{qMD} and ρ_{MD}

W_{qMD}	ρ_{MD}	Decision	Rule #
1.	1.	ADD 1 BED	217
	2.	ADD 1 BED	218
	3.	Goals Met	219
	4.	Goals Met	220
	5.	Goals Met	221
	6.	Goals Met	222
2.	1.	ADD 1 BED	223
	2.	ADD 1 BED	224
	3.	Goals Met	225
	4.	Goals Met	226
	5.	Goals Met	227
	6.	Goals Met	228
3.	1.	One variable heuristic	229
	2.	One variable heuristic	230
	3.	Goals Met	231
	4.	Goals Met	232
	5.	Goals Met	233
	6.	Goals Met	234
4.	1.	One variable heuristic	235
	2.	One variable heuristic	236
	3.	Goals Met	237
	4.	Goals Met	238
	5.	Goals Met	239
	6.	Goals Met	240
5.	1.	One variable heuristic	241
	2.	One variable heuristic	242
	3.	One variable heuristic?	243
	4.	One variable heuristic	244
	5.	One variable heuristic	245
	6.	One variable heuristic	246
6.	1.	One variable heuristic	247
	2.	One variable heuristic	248
	3.	One variable heuristic?	249
	4.	One variable heuristic	250
	5.	One variable heuristic	251
	6.	ADD 1 MD + 1 RN + 10 BEDS	252

Table 38: Possible Scenarios for W_{qMD} and ρ_{beds}

W_{qMD}	ρ_{beds}	Decision	Rule #
1.	1.	One variable heuristic	253
	2.	One variable heuristic	254
	3.	Goals Met	255
	4.	Goals Met	256
	5.	Goals Met	257
	6.	Goals Met	258
2.	1.	One variable heuristic	259
	2.	One variable heuristic	260
	3.	Goals Met	261
	4.	Goals Met	262
	5.	Goals Met	263
	6.	Goals Met	264
3.	1.	One variable heuristic	265
	2.	One variable heuristic	266
	3.	Goals Met	267
	4.	Goals Met	268
	5.	Goals Met	269
	6.	Goals Met	270
4.	1.	One variable heuristic	271
	2.	One variable heuristic	272
	3.	Goals Met	273
	4.	Goals Met	274
	5.	Goals Met	275
	6.	Goals Met	276
5.	1.	One variable heuristic	277
	2.	One variable heuristic	278
	3.	ADD 1 MD + 1 RN	279
	4.	ADD 1 MD + 1 RN	280
	5.	ADD 1 MD + 1 RN	281
	6.	ADD 1 MD + 1 RN	282
6.	1.	One variable heuristic	283
	2.	One variable heuristic	284
	3.	ADD 1 MD + 1 RN	285
	4.	ADD 1 MD + 1 RN	286
	5.	ADD 1 MD + 1 RN	287
	6.	ADD 1 MD + 1 RN	288

Table 39: Possible Scenarios for W_{qMD} and W_{qbeds}

W_{qMD}	W_{qbeds}	Decision	Rule #
1.	1.	Goals Met	289
	2.	Goals Met	290
	3.	Goals Met	291
	4.	Goals Met	292
	5.	ADD 1 MD + 2 RN +1 BED	293
	6.	ADD 1 MD + 3 RN +1 BED	294
2.	1.	Goals Met	295
	2.	Goals Met	296
	3.	Goals Met	297
	4.	Goals Met	298
	5.	ADD 1 MD + 2 RN +1 BED	299
	6.	ADD 1 MD + 4 RN +1 BED	300
3.	1.	Goals Met	301
	2.	Goals Met	302
	3.	Goals Met	303
	4.	Goals Met	304
	5.	ADD 1 MD + 2 RN +1 BED	305
	6.	ADD 1 MD + 4 RN +1 BED	306
4.	1.	Goals Met	307
	2.	Goals Met	308
	3.	Goals Met	309
	4.	Goals Met	310
	5.	ADD 1 MD + 3 RN +1 BED	311
	6.	ADD 1 MD + 4 RN +1 BED	312
5.	1.	ADD 1 MD	313
	2.	ADD 1 MD	314
	3.	ADD 1 MD	315
	4.	ADD 1 MD	316
	5.	ADD 1 MD + 2 RN +1 BED	317
	6.	ADD 1 MD + 3 RN +1 BED	318
6.	1.	ADD 1 MD	319
	2.	ADD 1 MD	320
	3.	ADD 1 MD	321
	4.	ADD 1 MD	322
	5.	ADD 1 MD + 2 RN +1 BED	323
	6.	ADD 1 MD + 3 RN +1 BED	324

Table 40: Possible Scenarios for W_{qbeds} and ρ_{RN}

W_{qbeds}	ρ_{RN}	Decision	Rule #
1.	1.	ADD 2 MD + 4 BEDS	325
	2.	ADD 2 MD + 2 BEDS	326
	3.	Goals Met	327
	4.	Goals Met	328
	5.	Goals Met	329
	6.	Goals Met	330
2.	1.	ADD 2 MD + 4 BEDS	331
	2.	ADD 2 MD + 2 BEDS	332
	3.	Goals Met	333
	4.	Goals Met	334
	5.	Goals Met	335
	6.	Goals Met	336
3.	1.	ADD 2 MD + 4 BEDS	337
	2.	ADD 2 MD + 2 BEDS	338
	3.	Goals Met	339
	4.	Goals Met	340
	5.	Goals Met	341
	6.	Goals Met	342
4.	1.	ADD 2 MD + 4 BEDS	343
	2.	ADD 2 MD + 2 BEDS	344
	3.	Goals Met	345
	4.	Goals Met	346
	5.	Goals Met	347
	6.	Goals Met	348
5.	1.	ADD 2 MD + 2 BEDS	349
	2.	ADD 2 MD + 2 BEDS	350
	3.	ADD 1 MD + 1 RN	351
	4.	ADD 1 MD + 1 RN	352
	5.	ADD 1 MD + 1 RN	353
	6.	ADD 1 MD + 1 RN	354
6.	1.	ADD 2 MD + 4 BEDS	355
	2.	ADD 2 MD + 2 BEDS	356
	3.	ADD 1 MD + 1 RN	357
	4.	ADD 1 MD + 1 RN	358
	5.	ADD 1 MD + 1 RN	359
	6.	ADD 1 MD + 1 RN	360

Table 41: Possible Scenarios for W_{qbeds} and ρ_{MD}

W_{qbeds}	ρ_{MD}	Decision	Rule #
1.	1.	ADD 3 RN + 3 BEDS	361
	2.	ADD 2 RN	362
	3.	Goals Met	363
	4.	Goals Met	364
	5.	Goals Met	365
	6.	Goals Met	366
2.	1.	ADD 3 RN + 3 BEDS	367
	2.	ADD 2 RN	368
	3.	Goals Met	369
	4.	Goals Met	370
	5.	Goals Met	371
	6.	Goals Met	372
3.	1.	ADD 3 RN + 3 BEDS	373
	2.	ADD 2 RN	374
	3.	Goals Met	375
	4.	Goals Met	376
	5.	Goals Met	377
	6.	Goals Met	378
4.	1.	ADD 3 RN + 3 BEDS	379
	2.	ADD 2 RN	380
	3.	Goals Met	381
	4.	Goals Met	382
	5.	Goals Met	383
	6.	Goals Met	384
5.	1.	ADD 2 RN + 2 BEDS	385
	2.	ADD 2 RN	386
	3.	ADD 1 RN + 1 BED	387
	4.	ADD 1 RN + 1 BED	388
	5.	ADD 1 RN + 1 BED	389
	6.	ADD 1 RN + 1 BED	390
6.	1.	ADD 3 RN + 3 BEDS	391
	2.	ADD 2 RN + 2 BEDS	392
	3.	ADD 3 RN + 3 BEDS	393
	4.	ADD 2 RN + 2 BEDS	394
	5.	ADD 3 RN + 1 BED	395
	6.	ADD 3 RN + 1 BED	396

Table 42: Possible Scenarios for W_{qbeds} and ρ_{beds}

W_{qbeds}	ρ_{beds}	Decision	Rule #
1.	1.	ADD 4 RN	397
	2.	ADD 3 RN	398
	3.	Goals Met	399
	4.	Goals Met	400
	5.	Goals Met	401
	6.	Goals Met	402
2.	1.	ADD 4 RN	403
	2.	ADD 3 RN	404
	3.	Goals Met	405
	4.	Goals Met	406
	5.	Goals Met	407
	6.	Goals Met	408
3.	1.	ADD 4 RN	409
	2.	ADD 3 RN	410
	3.	Goals Met	411
	4.	Goals Met	412
	5.	Goals Met	413
	6.	Goals Met	414
4.	1.	ADD 4 RN	415
	2.	ADD 3 RN	416
	3.	Goals Met	417
	4.	Goals Met	418
	5.	Goals Met	419
	6.	Goals Met	420
5.	1.	ADD 3 RN	421
	2.	ADD 2 RN	422
	3.	ADD 1 MD + 1 RN + 1 BED	423
	4.	ADD 1 MD + 1 RN + 1 BED	424
	5.	ADD 1 MD + 1 RN + 1 BED	425
	6.	ADD 1 MD + 1 RN + 1 BED	426
6.	1.	ADD 4 RN	427
	2.	ADD 1 MD + 2 RN + 1 BED	428
	3.	ADD 1 MD + 2 RN + 1 BED	429
	4.	ADD 1 MD + 2 RN + 1 BED	430
	5.	ADD 1 MD + 2 RN + 1 BED	431
	6.	ADD 1 MD + 2 RN + 1 BED	432

Table 43: Possible Scenarios for ρ_{RN} and ρ_{MD}

ρ_{RN}	ρ_{MD}	Decision	Rule #
1.	1.	ADD 8 BEDS	433
	2.	ADD 8 BEDS	434
	3.	ADD 8 BEDS	435
	4.	ADD 8 BEDS	436
	5.	ADD 8 BEDS	437
	6.	ADD 8 BEDS	438
2.	1.	ADD 5 BEDS	439
	2.	ADD 5 BEDS	440
	3.	ADD 5 BEDS	441
	4.	ADD 5 BEDS	442
	5.	ADD 5 BEDS	443
	6.	ADD 5 BEDS	444
3.	1.	ADD 2 BEDS	445
	2.	ADD 1 BED	446
	3.	Goals Met	447
	4.	Goals Met	448
	5.	Goals Met	449
	6.	Goals Met	450
4.	1.	ADD 2 BEDS	451
	2.	ADD 1 BED	452
	3.	Goals Met	453
	4.	Goals Met	454
	5.	Goals Met	455
	6.	Goals Met	456
5.	1.	ADD 2 BEDS	457
	2.	ADD 1 BED	458
	3.	Goals Met	459
	4.	Goals Met	460
	5.	Goals Met	461
	6.	Goals Met	462
6.	1.	ADD 2 BEDS	463
	2.	ADD 3 RN + 1 BED	464
	3.	Goals Met	465
	4.	Goals Met	466
	5.	Goals Met	467
	6.	Goals Met	468

Table 44: Possible Scenarios for ρ_{RN} and ρ_{beds}

ρ_{RN}	ρ_{beds}	Decision	Rule #
1.	1.	ADD 2 MD + 2 RN	469
	2.	ADD 2 MD + 2 RN	470
	3.	ADD 2 MD + 2 RN	471
	4.	ADD 2 MD + 2 RN	472
	5.	ADD 2 MD + 2 RN	473
	6.	ADD 2 MD + 2 RN	474
2.	1.	ADD 1 MD + 1 RN	475
	2.	ADD 1 MD + 1 RN	476
	3.	ADD 1 MD + 1 RN	477
	4.	ADD 1 MD + 1 RN	478
	5.	ADD 1 MD + 1 RN	479
	6.	ADD 1 MD + 1 RN	480
3.	1.	ADD 1 MD + 1 RN	481
	2.	ADD 1 MD + 1 RN	482
	3.	Goals Met	483
	4.	Goals Met	484
	5.	Goals Met	485
	6.	Goals Met	486
4.	1.	ADD 2 MD + 2 RN	487
	2.	ADD 1 MD + 1 RN	488
	3.	Goals Met	489
	4.	Goals Met	490
	5.	Goals Met	491
	6.	Goals Met	492
5.	1.	ADD 2 MD + 2 RN	493
	2.	ADD 1 MD + 1 RN	494
	3.	Goals Met	495
	4.	Goals Met	496
	5.	Goals Met	497
	6.	Goals Met	498
6.	1.	ADD 2 MD + 2 RN	499
	2.	ADD 1 MD + 1 RN	500
	3.	Goals Met	501
	4.	Goals Met	502
	5.	Goals Met	503
	6.	Goals Met	504

Table 45: Possible Scenarios for ρ_{MD} and ρ_{beds}

ρ_{MD}	ρ_{beds}	**Decision**	**Rule #**
1.	1.	ADD 4 RN	505
	2.	ADD 2 RN	506
	3.	ADD 1 BED	507
	4.	ADD 1 BED	508
	5.	ADD 1 BED	509
	6.	ADD 2 BEDS	510
2.	1.	ADD 4 RN	511
	2.	ADD 2 RN	512
	3.	ADD 1 BED	513
	4.	ADD 1 BED	514
	5.	ADD 1 BED	515
	6.	ADD 1 BED	516
3.	1.	ADD 4 RN	517
	2.	ADD 2 RN	518
	3.	Goals Met	519
	4.	Goals Met	520
	5.	Goals Met	521
	6.	Goals Met	522
4.	1.	ADD 4 RN	523
	2.	ADD 2 RN	524
	3.	Goals Met	525
	4.	Goals Met	526
	5.	Goals Met	527
	6.	Goals Met	528
5.	1.	ADD 4 RN	529
	2.	ADD 2 RN	530
	3.	Goals Met	531
	4.	Goals Met	532
	5.	Goals Met	533
	6.	Goals Met	534
6.	1.	ADD 4 RN	535
	2.	ADD 2 RN	536
	3.	Goals Met	537
	4.	Goals Met	538
	5.	Goals Met	539
	6.	Goals Met	540

Table 46: Scenarios for ρ_{RN}

ρ_{RN}	Decision	Rule#
1.	ADD 2 MD + 4 BEDS	1
2.	ADD 2 MD + 2 BEDS	2
3.	Goal met	3
4.	Goal met	4
5.	Goal met	5
6.	Goal over met	6

Table 47: Scenarios for ρ_{MD}

ρ_{MD}	Decision	Rule#
1.	ADD 2 BEDS	7
2.	ADD 1 BED	8
3.	Goal met?	9
4.	Goal met	10
5.	Goal met	11
6.	Goal over met	12

Table 48: Scenarios for ρ_{beds}

ρ_{beds}	Decision	Rule#
1.	ADD 4 RN	13
2.	ADD 2 RN	14
3.	Goal met?	15
4.	Goal met	16
5.	Goal met	17
6.	Goal over met	18

Table 49: Scenarios for W_{qRN}

W_{qRN}	Decision	Rule#
1.	Goal over met	19
2.	Goal met	20
3.	Goal met?	21
4.	Goal met	22
5.	ADD 1 MD + 1 RN	23
6.	ADD 2 MD + 2 RN	24

Table 50: Scenarios for W_{qMD}

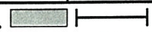
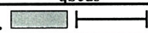
W_{qMD}	Decision	Rule#
1. 	Goal over met	25
2. 	Goal met	26
3. 	Goal met?	27
4. 	Goal met	28
5. 	ADD 1 MD	29
6. 	ADD 1 MD	30

Table 51: Scenarios for W_{qbeds}

W_{qbeds}	Decision	Rule#
1. 	Goal over met	31
2. 	Goal met	32
3. 	Goal met?	33
4. 	Goal met	34
5. 	ADD 2 RN	35
6. 	ADD 3 RN	36

APPENDIX F

F.1 Matrix of Changes to Parameters based on Rule Numbers (MVH)

Decision #	RN	MD	Beds	Decision #	RN	MD	Beds
1	0	2	0	43	0	0	1
2	0	1	0	44	0	0	1
3	999	999	999	45	999	999	999
4	999	999	999	46	999	999	999
5	999	999	999	47	999	999	999
6	999	999	999	48	999	999	999
7	0	1	0	49	1	0	3
8	0	1	0	50	1	0	2
9	999	999	999	51	999	999	999
10	999	999	999	52	999	999	999
11	999	999	999	53	999	999	999
12	999	999	999	54	999	999	999
13	0	1	0	55	1	0	3
14	0	1	0	56	1	0	2
15	999	999	999	57	999	999	999
16	999	999	999	58	999	999	999
17	999	999	999	59	999	999	999
18	999	999	999	60	999	999	999
19	0	1	0	61	1	0	3
20	0	1	0	62	1	0	0
21	999	999	999	63	1	0	0
22	999	999	999	64	1	0	0
23	999	999	999	65	1	0	0
24	999	999	999	66	1	0	0
25	-555	-555	-555	67	2	0	3
26	1	1	0	68	2	0	0
27	1	1	0	69	2	0	0
28	1	1	0	70	2	0	0
29	1	1	0	71	2	0	0
30	1	1	1	72	2	0	0
31	-555	-555	-555	73	4	0	0
32	1	1	0	74	2	0	0
33	1	1	0	75	999	999	999
34	1	1	0	76	999	999	999
35	1	1	0	77	999	999	999
36	1	1	1	78	999	999	999
37	0	0	2	79	4	0	0
38	0	0	1	80	2	0	0
39	999	999	999	81	999	999	999
40	999	999	999	82	999	999	999
41	999	999	999	83	999	999	999
42	999	999	999	84	999	999	999

Decision #	RN	MD	Beds	Decision #	RN	MD	Beds
85	4	0	0	134	1	1	0
86	2	0	0	135	1	1	0
87	999	999	999	136	1	1	0
88	999	999	999	137	1	1	0
89	999	999	999	138	1	1	0
90	999	999	999	139	3	1	0
91	4	0	0	140	2	1	0
92	2	0	0	141	2	1	0
93	999	999	999	142	2	1	0
94	999	999	999	143	2	1	0
95	999	999	999	144	3	1	0
96	999	999	999	145	999	999	999
97	4	0	0	146	999	999	999
98	2	0	0	147	999	999	999
99	1	1	0	148	999	999	999
100	1	1	0	149	2	0	0
101	1	1	0	150	2	0	1
102	1	1	0	151	999	999	999
103	4	0	0	152	999	999	999
104	4	1	0	153	999	999	999
105	4	1	0	154	999	999	999
106	4	1	0	155	1	0	1
107	4	1	0	156	2	1	0
108	4	1	0	157	999	999	999
109	999	999	999	158	999	999	999
110	999	999	999	159	999	999	999
111	999	999	999	160	999	999	999
112	999	999	999	161	2	1	0
113	0	1	0	162	3	2	0
114	0	1	0	163	999	999	999
115	999	999	999	164	999	999	999
116	999	999	999	165	999	999	999
117	999	999	999	166	999	999	999
118	999	999	999	167	1	0	2
119	1	1	0	168	2	0	2
120	1	1	0	169	1	0	0
121	999	999	999	170	1	0	0
122	999	999	999	171	1	0	0
123	999	999	999	172	1	0	0
124	999	999	999	173	1	0	1
125	1	1	0	174	2	1	1
126	1	1	0	175	2	0	0
127	999	999	999	176	2	0	0
128	999	999	999	177	2	0	0
129	999	999	999	178	2	0	0
130	999	999	999	179	1	1	0
131	1	1	0	180	2	0	0
132	1	1	0	181	0	1	10
133	1	0	0	182	0	1	7

Decision #	RN	MD	Beds	Decision #	RN	MD	Beds
183	999	999	999	232	999	999	999
184	999	999	999	233	999	999	999
185	999	999	999	234	999	999	999
186	999	999	999	235	-555	-555	-555
187	0	1	10	236	-555	-555	-555
188	0	1	7	237	999	999	999
189	999	999	999	238	999	999	999
190	999	999	999	239	999	999	999
191	999	999	999	240	999	999	999
192	999	999	999	241	-555	-555	-555
193	0	1	10	242	-555	-555	-555
194	0	1	7	243	-555	-555	-555
195	999	999	999	244	-555	-555	-555
196	999	999	999	245	-555	-555	-555
197	999	999	999	246	-555	-555	-555
198	999	999	999	247	-555	-555	-555
199	0	1	10	248	-555	-555	-555
200	0	1	7	249	-555	-555	-555
201	999	999	999	250	-555	-555	-555
202	999	999	999	251	-555	-555	-555
203	999	999	999	252	1	1	10
204	999	999	999	253	-555	-555	-555
205	0	1	10	254	-555	-555	-555
206	0	1	7	255	999	999	999
207	1	1	0	256	999	999	999
208	1	1	0	257	999	999	999
209	1	1	0	258	999	999	999
210	2	1	0	259	-555	-555	-555
211	0	1	10	260	-555	-555	-555
212	0	1	8	261	999	999	999
213	0	1	1	262	999	999	999
214	0	1	1	263	999	999	999
215	0	1	1	264	999	999	999
216	0	1	1	265	-555	-555	-555
217	0	0	1	266	-555	-555	-555
218	0	0	1	267	999	999	999
219	999	999	999	268	999	999	999
220	999	999	999	269	999	999	999
221	999	999	999	270	999	999	999
222	999	999	999	271	-555	-555	-555
223	0	0	1	272	-555	-555	-555
224	0	0	1	273	999	999	999
225	999	999	999	274	999	999	999
226	999	999	999	275	999	999	999
227	999	999	999	276	999	999	999
228	999	999	999	277	-555	-555	-555
229	-555	-555	-555	278	-555	-555	-555
230	-555	-555	-555	279	1	1	0
231	999	999	999	280	1	1	0

Decision #	RN	MD	Beds	Decision #	RN	MD	Beds
281	1	1	0	330	999	999	999
282	1	1	0	331	0	2	4
283	-555	-555	-555	332	0	2	2
284	-555	-555	-555	333	999	999	999
285	1	1	0	334	999	999	999
286	1	1	0	335	999	999	999
287	1	1	0	336	999	999	999
288	1	1	0	337	0	2	4
289	999	999	999	338	0	2	2
290	999	999	999	339	999	999	999
291	999	999	999	340	999	999	999
292	999	999	999	341	999	999	999
293	2	1	1	342	999	999	999
294	3	1	1	343	0	2	4
295	999	999	999	344	0	2	2
296	999	999	999	345	999	999	999
297	999	999	999	346	999	999	999
298	999	999	999	347	999	999	999
299	2	1	1	348	999	999	999
300	4	1	1	349	0	2	2
301	999	999	999	350	0	2	2
302	999	999	999	351	1	1	0
303	999	999	999	352	1	1	0
304	999	999	999	353	1	1	0
305	2	1	1	354	1	1	0
306	4	1	1	355	0	2	4
307	999	999	999	356	0	2	2
308	999	999	999	357	1	1	0
309	999	999	999	358	1	1	0
310	999	999	999	359	1	1	0
311	3	1	1	360	1	1	0
312	4	1	1	361	3	0	3
313	0	1	0	362	2	0	0
314	0	1	0	363	999	999	999
315	0	1	0	364	999	999	999
316	0	1	0	365	999	999	999
317	2	1	1	366	999	999	999
318	3	1	1	367	3	0	3
319	0	1	0	368	2	0	0
320	0	1	0	369	999	999	999
321	0	1	0	370	999	999	999
322	0	1	0	371	999	999	999
323	2	1	1	372	999	999	999
324	3	1	1	373	3	0	3
325	0	2	4	374	2	0	0
326	0	2	2	375	999	999	999
327	999	999	999	376	999	999	999
328	999	999	999	377	999	999	999
329	999	999	999	378	999	999	999

Decision #	RN	MD	Beds	Decision #	RN	MD	Beds
379	3	0	3	429	2	1	1
380	2	0	0	430	2	1	1
381	999	999	999	431	2	1	1
382	999	999	999	432	2	1	1
383	999	999	999	433	0	0	8
384	999	999	999	434	0	0	8
385	2	0	2	435	0	0	8
386	2	0	0	436	0	0	8
387	1	0	1	437	0	0	8
388	1	0	1	438	0	0	8
389	1	0	1	439	0	0	5
390	1	0	1	440	0	0	5
391	3	0	3	441	0	0	5
392	2	0	2	442	0	0	5
393	3	0	3	443	0	0	5
394	2	0	2	444	0	0	5
395	3	0	1	445	0	0	2
396	3	0	1	446	0	0	1
397	4	0	0	447	999	999	999
398	3	0	0	448	999	999	999
399	999	999	999	449	999	999	999
400	999	999	999	450	999	999	999
401	999	999	999	451	0	0	2
402	999	999	999	452	0	0	1
403	4	0	0	453	999	999	999
404	3	0	0	454	999	999	999
405	999	999	999	455	999	999	999
406	999	999	999	456	999	999	999
407	999	999	999	457	0	0	2
408	999	999	999	458	0	0	1
409	4	0	0	459	999	999	999
410	3	0	0	460	999	999	999
411	999	999	999	461	999	999	999
412	999	999	999	462	999	999	999
413	999	999	999	463	0	0	2
414	999	999	999	464	3	0	1
415	4	0	0	465	999	999	999
416	3	0	0	466	999	999	999
417	999	999	999	467	999	999	999
418	999	999	999	468	999	999	999
419	999	999	999	469	2	2	0
420	999	999	999	470	2	2	0
421	3	0	0	471	2	2	0
422	2	0	0	472	2	2	0
423	1	1	1	473	2	2	0
424	1	1	1	474	2	2	0
425	1	1	1	475	1	1	0
426	1	1	1	476	1	1	0
427	4	0	0	477	1	1	0
428	2	1	1	478	1	1	0

Decision #	RN	MD	Beds	Decision #	RN	MD	Beds
479	1	1	0	510	999	999	999
480	1	1	0	511	4	0	0
481	1	1	0	512	2	0	0
482	1	1	0	513	0	0	1
483	999	999	999	514	0	0	1
484	999	999	999	515	0	0	1
485	999	999	999	516	0	0	1
486	999	999	999	517	4	0	0
487	2	2	0	518	2	0	0
488	1	1	0	519	999	999	999
489	999	999	999	520	999	999	999
490	999	999	999	521	999	999	999
491	999	999	999	522	999	999	999
492	999	999	999	523	4	0	0
493	2	2	0	524	2	0	0
494	1	1	0	525	999	999	999
495	999	999	999	526	999	999	999
496	999	999	999	527	999	999	999
497	999	999	999	528	999	999	999
498	999	999	999	529	4	0	0
499	2	2	0	530	2	0	0
500	1	1	0	531	999	999	999
501	999	999	999	532	999	999	999
502	999	999	999	533	999	999	999
503	999	999	999	534	999	999	999
504	999	999	999	535	4	0	0
505	4	0	0	536	2	0	0
506	2	0	0	537	999	999	999
507	999	999	999	538	999	999	999
508	999	999	999	539	999	999	999
509	999	999	999	540	999	999	999

* 999 indicates the goals are met

-555 indicates OVH is required

F.2 Matrix of Changes to Parameters based on Rule Numbers (OVH)

Decision #	RN	MD	Beds
1	0	2	4
2	0	1	2
3	999	999	999
4	999	999	999
`5	999	999	999
6	999	999	999
7	0	0	2
8	0	0	1
9	999	999	999
10	999	999	999
11	999	999	999
12	999	999	999
13	4	0	0
14	2	0	0
15	999	999	999
16	999	999	999
17	999	999	999
18	999	999	999
19	999	999	999
20	999	999	999
21	999	999	999
22	999	999	999
23	1	1	0
24	2	2	0
25	999	999	999
26	999	999	999
27	999	999	999
28	999	999	999
29	0	1	0
30	0	1	0
31	999	999	999
32	999	999	999
33	999	999	999
34	999	999	999
35	2	0	0
36	3	0	0

* 999 indicates the goals are met

-555 indicates OVH is required