Hospital Ward and ICU Flow Simulation Using Discrete-Event Modeling

University of Toronto: Faculty of Applied Science and Engineering

Victoria Piroian

Table of Contents

1.0 Introduction

This report aims to investigate further the performance of various operational strategies taken to reduce waiting times at a large Canadian hospital. Simulation in Simio will be used to further understand waiting and service times across the separate wards of the hospital, including intensive and acute care. The document will begin with a description of the input modelling, which outlines how the team used the given hospital data including arrival data used for bed requests, the acute service times, and the ICU service times to determine a model for arrival times into the system, and estimated distributions for the service times at each ward using Microsoft Excel and R. A base model for the system is then created including all the possible patient flows in Simio, including some performance measures such as waiting times across the wards, bed utilizations, and others. Finally, a scenario analysis is conducted first between two alternatives, and then comparing multiple alternatives.

2.0 Input Modelling

2.1 Arrivals

In order to estimate a model for bed request arrivals to acute care, the team used the ArrivalData.xlsx. This file contains the number of arrivals to the system for each hour of the day across 50 weeks. Using this data in Excel, the team created a pivot table to find the average rate of acute bed requests by the hour and day of the week. The rows in the pivot table were the day of the week and hours of the day, and the columns were the average number of arrivals. A bar chart was then created from the pivot table (see Figure 1). These arrival rates will later be used in the rate tables in Simio to simulate arrivals for acute bed requests.

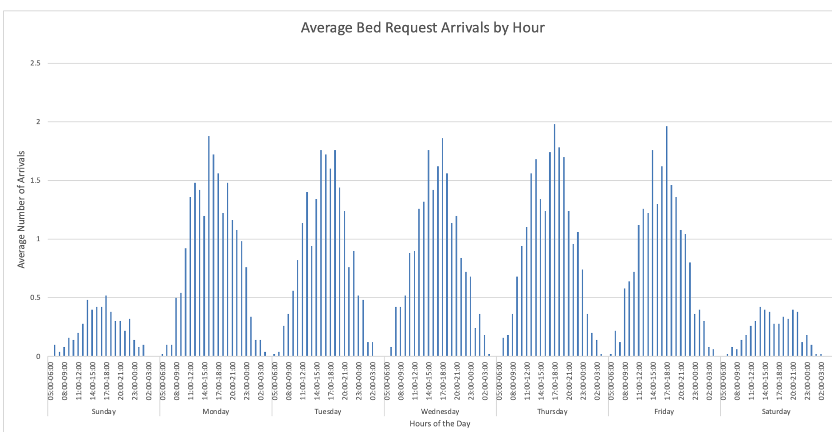


Figure 1: The estimated average number of bed requests by the hour and day of the week

2.2 Acute Treatment Times

R was used to determine the treatment times for the acute awards. The provided data on acute times was first sorted in excel by acute category, and whether the patient had crashed or not. This resulted in a crash and no-crash set of data for each of the four acute categories for a total of 8 datasets. Each dataset was then passed to R where it was fit to normal, lognormal, weibull, gamma, and exponential distributions using the *fitdist()* function from the *fitdistrplus* library. Next the QQ plots for each of the distributions were plotted together in order to gain an understanding on what distribution best fit the data. This was done 8 times; 4 times for each acute category when a crash resulted, and 4 more times for each acute category when no crash occurred (see Appendix A for the R code and its output for the distribution of service times for non-crashing cardiovascular patients. The QQ plots for each scenario can be seen below (see Appendix B).

Based on the QQ plots generated, the team selected the distributions that best fit the model, and found the parameters for each distribution using the *kstest* value in the *gofstat()* function. This resulted in the selection of the distributions for each of the acute scenarios outlined in Appendix C.

2.3 ICU Treatment Times

The process for determining the ICU service times for patients was the same as the treatment time for the acute wards. First, the data was divided into two scenarios: (1) the patient died at the ICU, and (2) the patient survived the ICU. Next, the data was passed through the R code and fit to normal, lognormal, weibull, gamma, and exponential distributions, similar to the acute data. The QQ plots for each of the distributions were then plotted (see Appendix B). Similarly, the QQ plots were used to select the best-fitting distributions, and the parameters were found using the *kstest* value in the *gofstat()* function in R. The optimal distributions are outlined in Appendix C.

2.4 Implementation and Parameterization

After obtaining the arrival model, input distributions, and the parameter estimations using the data, the team was then able to implement these changes into the base model in Simio (see section 3.0 for the full breakdown of the base model). In order to implement the arrival model from section 2.1, the team used the rates by the hour for each day of the week and placed this data table into the Rate Tables under the Data tab for the base model. This specified the arrivals of the patients (ModelEntity) into the system based on the hourly arrival model. In order to implement the chosen distributions and their parameters from sections 2.2-2.3 into Simio, each of the Processing Time properties for each of the servers in Simio contains two different distributions, one for if the patient is crashing and is sent to the ICU, and the other for if the patient is not crashing and can exit the system afterward. For the server in the ICU, one distribution would represent the patient dying, and the other for if the patient survives. Furthermore, in the case where the service time is exponential or gamma, the parameter would be one divided by the rate outputted from R, rather than just the rate.

3.0 Base Model

The sections that follow will provide insight into the implementation of the given scenario and its related data into Simio as well as the design of experiments and estimation of performance measures.

3.1 Logic

The Simio model is composed of several primary components including the acute and ICU wards (the servers), the patients traveling through the system (the ModelEntities), the regular bed request and external admissions into the system (the sources), the discharge, alternative hospital, and death sinks, as well as the connectors that enable patient flow throughout the system (see Figure 2 for an overview of the base model).

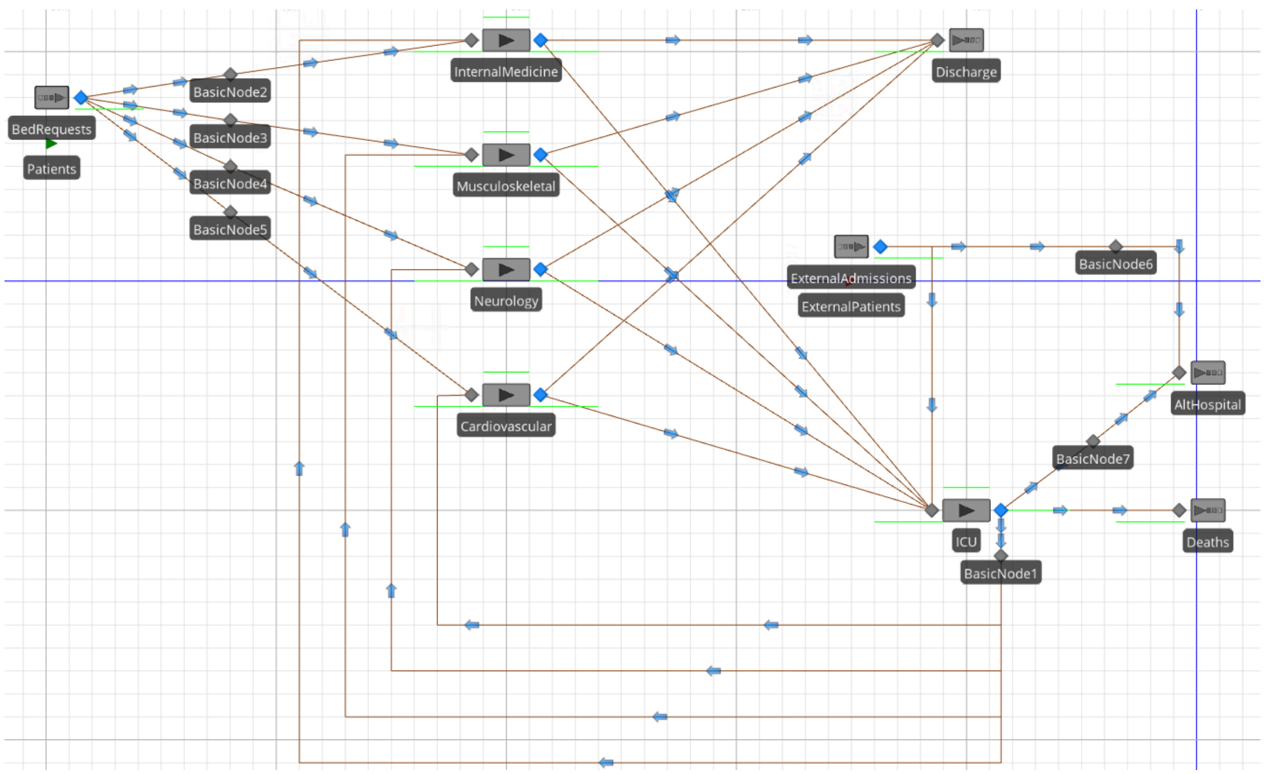


Figure 2: Overview of the completed base model in Simio

First, four servers were created to represent each of the acute wards (InternalMedicine, Musculoskeletal, Neurology, Cardiovascular), each with its own number of beds which are reflected in the Initial Capacity properties for the servers. Once a patient requests a bed in one of the acute wards, they are either admitted into that server if there is space (the number of ModelEntities being serviced < Initial Capacity), or the patient waits in the Input Buffer of that server which has a capacity of infinity. In order to assign each ModelEntity as a patient requesting a bed specifically for one of the acute wards, the team used the AcuteTimes dataset to see the proportions of patients that are requesting a bed in each of the wards. It was found that 40% of patients were requesting a bed in InternalMedicine, 30% for Musculoskeletal, 20% for Neurology, and 10% for Cardiovascular. These proportions were used as the Selection Weights for the connectors from the BedRequests source to their designated ward. There is also a basic node in between the source and the ward servers which assigns each patient a model entity state called OriginalWardId, which keeps track of the original ward each patient requested a bed for. For example, each patient entering InternalMedicine from BedRequests would be assigned OriginalWardId = 1. Once at their acute ward, the patient then spends a random amount of time being serviced according to the distributions found in section 2.2. The distribution not only depends on which acute ward they are in but also if that patient will be going to the ICU after their acute treatment (crash or not crash). An example of the processing time for a patient in Neurology is *Math.If(ModelEntity.ProbICU == 1, Random.Weibull(0.987769, 2.184903), Random.Lognormal(2.096512, 0.512613))*.

The team has implemented several processes (one for each acute ward) in order to move 15% of patients finishing their acute care to receive more treatment in the ICU (the other 85% move towards the Discharge sink where they leave the system). Upon entering each of the acute wards, there is an Add-On process trigger for the Input_*AcuteWardName*_Entered. This process then activates a Decide step with the following condition: *Random.Uniform(0, 1) <= Math.Max(0.15 - (ModelEntity.AcuteReturn * 0.05), 0)*. If this condition is true, then the patient has crashed and must head to the ICU (assigns the ProbICU model entity state a value of 1), otherwise, they exit the system and are assigned ProbICU = 0. This decision also considers the fact that every time a patient in the system crashes and heads to the ICU, their probability of returning to the ICU again is decreased by 5%. If the ICU has beds available (the full capacity is not being used), then the crashing patient is transferred to the ICU server, otherwise, they continue to occupy a bed in the acute ward which is implemented in Simio through the acute ward having an output buffer capacity of 0 and the ICU having an input buffer capacity of 0.

The ICU has 16 beds, so this server's Initial Capacity is 16. Patients in the ICU spend a random amount of time being serviced according to the distributions found in section 2.3 and is as follows: *Math.If(ModelEntity.ProbDeath == 1, Random.Gamma(0.9960113, 1/0.6881215), Random.Gamma(0.9991930, 1/0.1918584))*. This processing time is dependent on the ModelEntity state ProbDeath (whether the patient dies or not). Using the ICU historical data, it was found that 5% of patients pass away in the ICU, so the following state assignment condition is implemented in the input buffer to assign each model entity a ProbDeath: *Math.If(Random.Uniform(0, 1) <= 0.05, 1, 0)*. After this ICU treatment, patients who did not pass away return to their original acute ward (if there is space) based on the OriginalWardId which was assigned at the basic nodes after entering the system for the first time. In the case where there is no space in their original acute ward, the patient is diverted to the acute ward with the most space, all of which is implemented through the *ICUOutput* process. This process splits into 5 different Decision nodes based on the original ward of the model entity entering this process, one representing each of the OriginalWardId's for acute wards, and the fifth Decision node is for the case when the patient entered the system through the ExternalAdmission source (represented by OriginalWardId = 6). Note that if this is not the patient's first time entering the ICU, the ward ID being checked would be their most recent ward, not the original one. After identifying the original ward of each patient, the capacity of their original ward is checked to see if there is space for them and if so, they return to that server. Otherwise, the ModelEntity reaches more Decision nodes where many comparisons between the remaining capacities are performed to determine the acute ward with the highest number of beds available. Alternatively, if the patient passed away, their ModelEntity is deferred to the Deaths sink. In each of these cases, the patient is assigned a new ModelEntity state called NewWard which represents the server/sink they will be directed to after leaving the ICU this time. In the case where all the acute wards are full, the patient is assigned a NewWard = 5, meaning they leave the system and are directed to the AltHospital sink. Based on each ModelEntity's NewWard and ProbDeath, they are directed to either of the acute wards, AltHospital, or the Deaths sink.

Patients arriving from the ExternalAdmissions source have a random exponential interarrival time: *Random.Exponential(48 hours)*, and are directed to the ICU as long as its remaining capacity is above 0, otherwise, they are deferred to the AltHospital sink.

3.2 Design of Experiments

A warmup period was introduced into the Simio base model and was also implemented into the rest of the scenarios in order to reduce the bias of the results. Doing so allows the simulation to begin collecting results after the model exhibits conditions that are typical in steady-state. In order to determine the exact warmup period, the team plotted the number of patients in the system for the first year of operation. From this original graph, it was evident that the average number of patients in the system began to flatten much earlier on in the year, and so many more graphs were created, but over progressively smaller time periods. As can be seen in Figure 3, the average number of patients in the system begins to flatten around 12/28/2022. If the system starts on 11/28/2022 and the average number in the system stabilizes around 12/28/2022, then we can estimate the warmup period for the model to be approximately 30 days. Since the run length is typically ten times the warmup period, the run length for this model is 300 days.

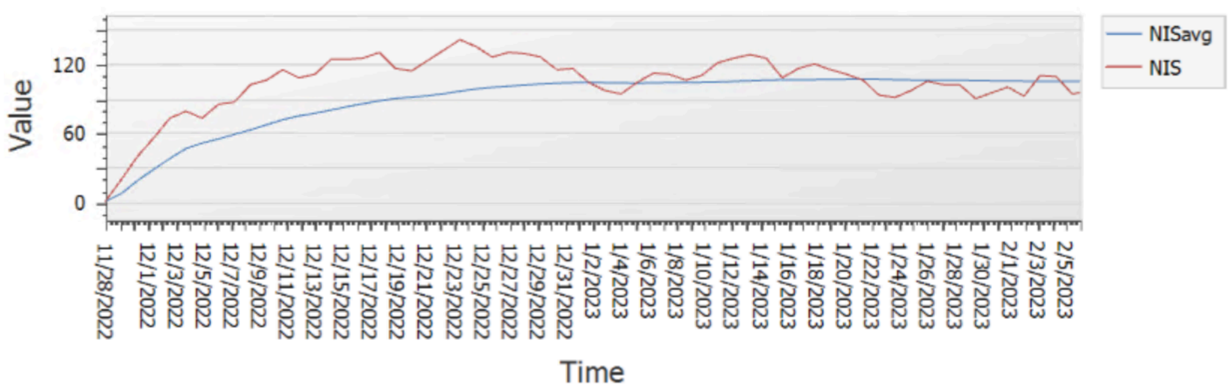


Figure 3: The average number of patients in the system plotted for the first two months of the simulation

3.3 Performance Measures

The Average Waiting Time and Queue Lengths were found for each of the acute wards, and the Bed Utilizations were found for each of the acute wards and the ICU, all using their corresponding methods. These performance measures can be found in Appendix D. In order to determine the response that measures the long-run proportion of external patients that are turned away because the ICU is full and the proportion that have to leave the system because acute care is full, the team implemented tally statistics and new ModelEntity states. For the first of these performance measures, the team created the new *ExternalRejection* model entity state which is set to one for patients who go from the *ExternalAdmissions* source and enter the *AltHosposital* sink because the ICU is full (implemented at *BasicNode6*). Upon entering the *AltHospital* sink, the *InputAltHospital_Entered* process is triggered. This process then has a Decision node that splits on the value of *ExternalRejection*. If the user came from *ExternalAdmission*, then the

ExtPatientsTurnedAway tally statistic is increased by the value of *ExternalRejection*, then divided by the total number of patients entering the *AltHospital* sink to find the proportion of external patients rejected by the ICU. Similarly, the proportion of ICU patients who are rejected from entering acute care and therefore are directed to the *AltHospital* is found using the model entity state *AcuteFullPatientLeaves* and the same process for the previous response. When a patient leaves the ICU and has a NewWard = 5 (meaning all the acute wards are full, see section 3.1), they are directed to the *AltHospital* sink. The entities moving along this connector reach *BasicNode7* where the *AcuteFullPatientLeaves* is set to one and is summed in the *PatientsSentToAltHospital* tally statistic. This statistic is then divided by the number of patients exiting the ICU to find the proportion of ICU patients who were rejected from acute care.

The average number of acute beds that are blocked is found using the *BedBlocked* model entity state which is assigned a value of one in any of the acute servers after being processed if they have to wait in that bed because the ICU is full. This is implemented by assigning *BedBlocked* the following value: *Math.If(ICU.Capacity.Remaining == 0, 1, 0)*. Note that since the input buffer of the ICU and the output buffer of the acute wards are zero, the team determined if a patient was blocking a bed by checking if the patient was still at the server even after their processing time was completed. The *BlockedBedCount* tally statistic was then calculated as the sum of *BedBlocked* at the input buffer of the ICU. Finally, the *AverageBedsBlocked* response was calculated using the *BlockedBedCount*'s number of observations. The resulting values for all of these responses as well as their half-widths around found in Appendix E.

The results describe the average waiting times and queue lengths for all four acute wards. These values show that the neurology ward has the longest wait times as well as the longest queue lengths, which have half-widths of 0.0967 and 0.3336 respectively. Looking at the bed utilizations it can be seen that the neurology ward also has the highest bed utilization of 85.9804 with a half-width of 0.9008, followed by ICU with a bed utilization of 85.0242 and a half-width of 1.0793. The remaining responses provide information on the issues related to hospital capacity in the ICU and acute wards. There were roughly 50 observations of external patients who were turned away from the ICU, and around 232 patients were sent from the ICU to an alternative hospital as the acute wards are full. On average, there were 246 beds blocked in the ICU across the full run length of the system due to patients occupying acute beds even after they have been serviced. These observations will be further analyzed and improved in the following sections using scenario analysis.

4.0 Scenario Analysis

The following section will discuss the variations from the base model in two different scenarios, the first of which combines all four acute wards into one larger server, and the second of which keeps the acute wards separate but turns patients away from them when the queue length of the arrivals exceeds a certain threshold T.

4.1 Comparing Two Scenarios

The combination of all four acute care wards into one larger, non-specialized ward resulted in changes to the patient flow of the model. The following section will detail the modifications to conditional paths that patients move along, the implementation of new nodes, as well as the results obtained from this scenario, and how it compares to the base model.

4.1.1 Explanation

The major change in this model was combining the four acute wards into one larger ward as depicted in Figure 4. Doing so would make it more difficult for the team to see the different categories of patients entering the single acute ward. To avoid this confusion the team added nodes to assign a number to each patient that would identify which ward they belonged to. This was done before they arrived at the acute ward. The new server had an increased capacity of 119 patients (the sum of all the previous acute ward capacities). When entering this server, patients were labeled as ICU or non-ICU patients as well as assigned the processing time distribution according to their category of illness. The processing time distributions are set as a model entity state for each patient entity, the two steps above are implemented through processes with conditional statements based on each model entity's states. As the processing time in this model is 10% more than the result of the distribution, the team set the processing time to 1.1 times the model entity state variable for processing time. Leaving the acute ward, there are only two branches unlike the base model, one leads to the *Discharge* sink where patient entities are destroyed, and the other to the ICU on the condition of what they had been previously labeled as. The ICU patients are then classified as dead or not dead and processed accordingly. If they are deceased they go to the *Deaths* sink to be destroyed, or if there is space in the *Acute* server they will return to the acute wards, otherwise be directed to *AltHospital*. Similar to the base model, this scenario also contains the *ExternalPatients* model entity that is sent to ICU, capacity permitting, or otherwise sent to *AltHospital*. This scenario measures the same responses as the base model, though results in only one response for the acute ward. This results in one average waiting time, queue length, and average bed utilization response respectively, as described in the next section.

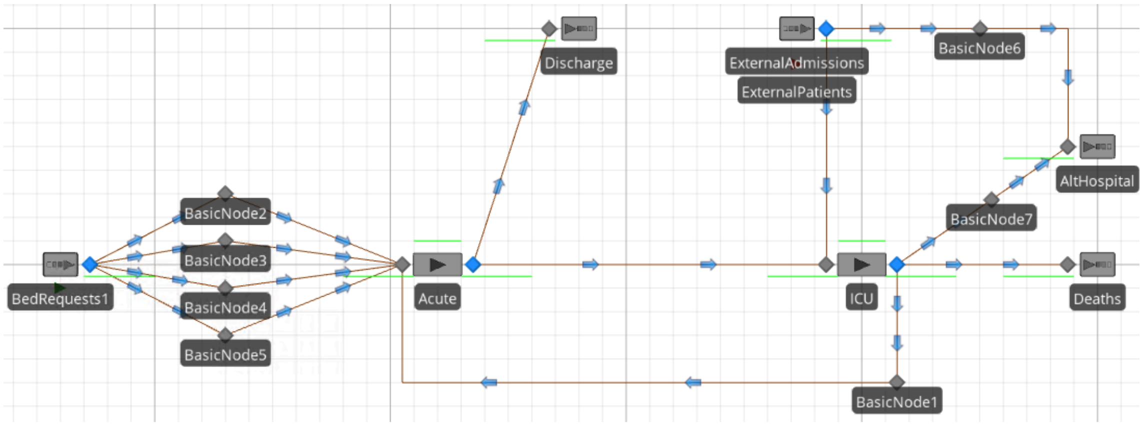


Figure 4: Overview of the Scenario 1 modification of the base model

4.1.2 Results

The team conducted an experiment for this scenario with 40 replications, a warm-up period of 30 days, and a total run length of 300 days. These values were chosen to be the same as that of the base model's experiment in order to allow for easier comparison between the two alternatives. When comparing the response values in Appendix F, it can be observed that the base model's utilization of ICU beds is slightly lower than this new model's meaning that the new scenario utilizes the bed in the ICU for a larger proportion of the time than the base model. In addition, we can see that the average waiting times, queue lengths, and bed utilizations are all much higher in scenario 1 than in the base model. The increase in queue lengths and average waiting times suggests that despite this model's higher utilization of servers, it would be worse than the base model due to the increase in queue lengths and longer patient waiting times. Referring to the remaining responses in Appendix F, it can be seen that the proportion of external patients turned away, and the proportion of ICU patients that have blocked beds has increased from the base model to scenario 1.

We can also use Appendix F to compare the variance of the responses in both models by observing any changes in the half-widths (see Appendix E). It can be seen, for example, that for the last three responses the averages and half-widths have generally increased. The proportion of external patients that were turned away has increased as well as the average number of ICU beds blocked. The number of beds blocked also has a larger half-width than in the base model, meaning there is a larger variation in the average value. This would result in even larger wait times after processing in the acute ward during the busiest times at the hospital.

After merging the acute wards, we see increased wait times, queue lengths, and in the proportions of patients that have been turned away due to a lack of beds. Through this comparison and analysis, the team has decided not to recommend this model over the original base model.

4.2 Comparing Multiple Scenarios

To help the hospital determine when arrivals should be allowed to enter the acute wards, and when they should be diverted to another hospital, an optimal policy for diverting arrivals was created. This policy checks when the total queue length for the acute care wards exceeds a critical threshold of value T and diverts arrivals in order to diminish congestion.

4.2.1 Explanation

To implement this new logic in the hospital model, a new connector was created from the *BedRequests* source node to the *AltHospital* sink node, and a new standard property named *Divert_Arrivals_Threshold* was created that would store the chosen T values. Selection weights of all the connectors coming out of the *BedRequests* source node were then adjusted so that the new connection would be accounted for. This was done by setting the selection weight of the new connection to the sum of all the patients waiting in the acute input buffers, and ensuring that this value was less than the new standard property:

$(InternalMedicine.AllocationQueue.NumberWaiting\ +\ Musculoskeletal.AllocationQueue.NumberWaiting\ +\ Neurology.AllocationQueue.NumberWaiting\ +\ Cardiovascular.AllocationQueue.NumberWaiting)\ >\ Divert\_Arrivals\_Threshold.$ All the connectors leading to the acute wards were then adjusted to check that the previous selection weight was being met while ensuring that the total queue length was less than T. This was done before admitting patients into the hospital. Finally, a new ModelEntity state for marking diverted patients was created and tallied in order to count and calculate the amount and proportion of patients being diverted.

With the base model adjusted to account for diverted patients, the new simulation model was used to find the optimal threshold value that minimized the objective function compromising of the weighted sum of the expected average total queue length and proportion of diverted patients:

$$Min\text{: } Long\ run\ average\ queue\ length\ +\ (120\ \times\ long\ run\ proportion\ of\ diverted\ arrivals)$$

4.2.2 Results

To find the ideal T value that minimized the objective function, a new experiment was created for this scenario. By creating a new primary response containing the objective function stated above, and passing it the “minimize” parameter, the team was able to use Kim and Nelson

Ranking (KN) to compare multiple T values simultaneously. A range of T's from 1 to 15 were selected with a minimum replication amount of 10, and a limit of 250 replications for each scenario. The warm-up period and run length of each replication were kept at 30 days and 300 days respectively, consistent with the other models. Finally, in order to prevent the experiment from running too long in the event of near ties, an indifference zone of 0.75 was selected for this experiment, which represented the smallest difference that was worth detecting. Given that the objective values were in the range of [29, 38], it was small enough to be acceptable while limiting the computation time needed.

Running the experiment produced the results found in Appendix G, where it is evident that a threshold value (T) of 7 is optimal for minimizing the objective function. This means that when 7 patients are waiting to be served in the queue of the acute wards, new patients will be directed to another hospital to prevent congestion.

The small half-width (see Appendix E & Appendix G) reported for the selected T value also suggests that there is a small margin of error in the 95% confidence interval that was picked. It is also demonstrated that T = 6 & T = 8 are close alternatives to the optimal value found since they result in similar objective values and have low half-width values as well. Overall, implementing a diversion policy also significantly reduces the number of beds that are blocked in the system which is overall a positive outcome and will ensure a more efficient treatment in the hospital as a whole.

5.0 Discussion

Through exploring the various models discussed above, the team suggests that the hospital implements a model that involves separate specialized wards. As discussed above, the wait times and queue lengths increase when the four acute wards are merged into 1. In addition, the number of beds blocked by patients increases when there is only one acute ward. This being considered, it is the team's recommendation that the hospital not adopt the single acute care ward model. Furthermore, the hospital should implement a diversion policy. This will reduce the congestion of the system and allow for more efficient treatment of patients. This is depicted in section 4.2, where the average count of beds blocked is drastically decreased and the wait times, as well as queue lengths, are much smaller. This leads the team to recommend that the hospital adopt the diversion policy for cases when arrivals occur at a time when there are 7 or more patients waiting for acute ward beds.

6.0 Appendices

Appendix A:

FitDistExample.R*

Source on Save

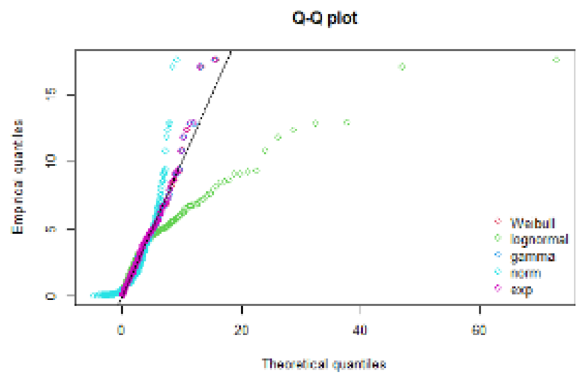
```r
# This file uses the fitdistrplus package. You'll need to install it first
# by running the command: 'install.packages("fitdistrplus")' in R
require(fitdistrplus)


my_log <- file("cardio0.txt") # File name of output log

sink(my_log, append = TRUE, type = "output") # Writing console output to log file
sink(my_log, append = TRUE, type = "message")

cat(readChar(rstudioapi::getSourceEditorContext()$path, # Writing currently opened R script to file
             file.info(rstudioapi::getSourceEditorContext()$path)$size))


# import the data from .csv file
AcuteTimes <- read.csv("Acute.csv", header=TRUE,
                       sep=",", na.strings=c("NA", ""), stringsAsFactors=FALSE, as.is=TRUE)

plotdist(AcuteTimes$Acute.Time..Days., histo = TRUE, demp = TRUE)


# fit a lognormal distribution to the LOS samples
fln <- fitdist(AcuteTimes$Acute.Time..Days., "lnorm")
summary(fln)
plot(fln)

# fit a weibull distribution
fw <- fitdist(AcuteTimes$Acute.Time..Days., "weibull")
summary(fw)
plot(fw)

# fit a gamma distribution
fg <- fitdist(AcuteTimes$Acute.Time..Days., "gamma")
summary(fg)
plot(fg)

fn <- fitdist(AcuteTimes$Acute.Time..Days., "norm")
summary(fn)
plot(fn)

fe <- fitdist(AcuteTimes$Acute.Time..Days., "exp")
summary(fe)
plot(fe)

# compare the fits
par(mfrow = c(2, 2))
plot.legend <- c("Weibull", "lognormal", "gamma", "norm", "exp")
denscomp(list(fw, fln, fg, fn, fe), legendtext = plot.legend)
qqcomp(list(fw, fln, fg, fn, fe), legendtext = plot.legend)
cdfcomp(list(fw, fln, fg, fn, fe), legendtext = plot.legend)

# Perform GofFit tests
gof_results <- gofstat(list(fw, fln, fg, fn, fe), fitnames = c("weibull", "lnorm", "gamma", "norm", "exp"))
gof_results
gof_results$kstest

gof_results$chisq
gof_results$chisqpvalue
gof_results$chisqtable
```

Figure 5: R code used in finding the distribution of service times for cardiovascular acute ward for non-crashing patients (CardioVascular0)

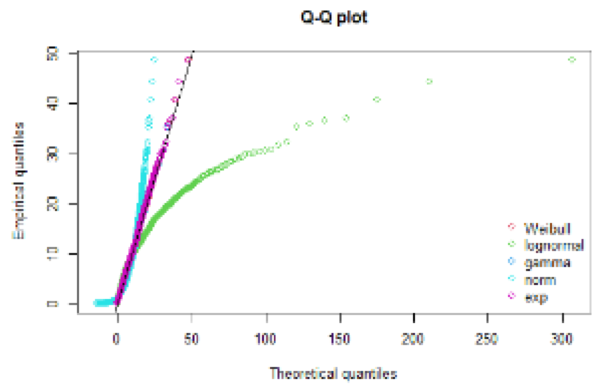
```
Fitting of the distribution ' lnorm ' by maximum likelihood
Parameters :
         estimate  Std. Error
meanlog 1.3667820 0.008800966
sdlog   0.2565901 0.006222798
Loglikelihood:  -1211.628   AIC:  2427.256   BIC:  2436.747
Correlation matrix:
        meanlog sdlog
meanlog       1     0
sdlog         0     1

Fitting of the distribution ' weibull ' by maximum likelihood
Parameters :
      estimate Std. Error
shape 4.557665  0.1175130
scale 4.424088  0.0351619
Loglikelihood:  -1188.401   AIC:  2380.801   BIC:  2390.292
Correlation matrix:
          shape     scale
shape 1.0000000 0.3214581
scale 0.3214581 1.0000000

Fitting of the distribution ' gamma ' by maximum likelihood
Parameters :
       estimate Std. Error
shape 16.244519  0.7800176
rate   4.014195  0.1957545
Loglikelihood:  -1191.822   AIC:  2387.643   BIC:  2397.134
Correlation matrix:
          shape      rate
shape 1.0000000 0.9846552
rate  0.9846552 1.0000000

Fitting of the distribution ' norm ' by maximum likelihood
Parameters :
     estimate Std. Error
mean 4.046612 0.03327628
sd   0.970162 0.02352977
Loglikelihood:  -1180.349   AIC:  2364.699   BIC:  2374.189
Correlation matrix:
     mean sd
mean    1  0
sd      0  1

Goodness-of-fit statistics
                                weibull     lnorm      gamma
Kolmogorov-Smirnov statistic 0.03653112 0.0684319 0.05205845
Cramer-von Mises statistic   0.17493045 0.9854658 0.49675477
Anderson-Darling statistic   1.37627995 5.7528736 2.78215647
                                   norm
Kolmogorov-Smirnov statistic 0.02178902
Cramer-von Mises statistic   0.06662196
Anderson-Darling statistic   0.38359791

Goodness-of-fit criteria
                                weibull    lnorm    gamma
Akaike's Information Criterion 2380.801 2427.256 2387.643
Bayesian Information Criterion 2390.292 2436.747 2397.134
                                   norm
Akaike's Information Criterion 2364.699
Bayesian Information Criterion 2374.189
       weibull          lnorm           gamma            norm
"not rejected"     "rejected"      "rejected" "not rejected"
 weibull    lnorm    gamma     norm
44.21895 80.48655 53.52692 33.59121
     weibull        lnorm         gamma          norm
4.947792e-03 2.647914e-08 3.107254e-04 7.130458e-02
         obscounts theo weibull theo lnorm theo gamma theo norm
<= 2.359        33     47.03666   20.18188   23.86883  34.82627
<= 2.635        33     29.42433   31.23006   29.85683  27.08002
<= 2.87         33     33.94205   43.49395   40.03437  33.80652
<= 3.045        33     31.18703   42.62054   39.08895  32.58485
<= 3.189        33     29.62105   40.82568   37.75886  31.80096
<= 3.34         33     34.90849   47.25491   44.30838  38.12267
<= 3.43         33     22.63101   29.77806   28.30809  24.92616
<= 3.553        33     33.01567   41.96734   40.44352  36.48390
<= 3.634        33     22.96878   28.09140   27.44253  25.38073
<= 3.737        33     30.47457   35.84130   35.45490  33.57850
<= 3.896        33     49.39212   54.68751   55.05975  53.97677
<= 3.995        34     31.85701   33.14091   33.92905  34.40130
<= 4.077        33     26.81616   26.68030   27.61757  28.65123
<= 4.201        33     40.92689   38.72136   40.54703  43.11658
<= 4.284        33     27.40085   24.67353   26.11153  28.41731
<= 4.363        33     25.89679   22.47124   23.95529  26.50414
<= 4.438        33     24.27731   20.37289   21.85179  24.52217
<= 4.561        33     38.81734   31.31299   33.80544  38.53382
<= 4.654        33     28.23693   21.91381   23.78937  27.49646
<= 4.757        34     29.85756   22.49859   24.50465  28.57941
<= 4.851        33     25.72966   18.93142   20.65393  24.22159
<= 5.029        33     44.14300   31.81609   34.68582  40.70733
<= 5.244        33     44.42326   31.88998   34.52603  40.00409
<= 5.511        33     41.11395   30.89678   32.83361  36.52156
<= 5.971        33     39.07128   35.54722   35.97906  35.65176
> 5.971         23     16.83024   43.16025   33.58482  20.10392
```

Figure 6: Output from the R code for the service time of cardiovascular acute ward for non-crashing patients (CardioVascular0)

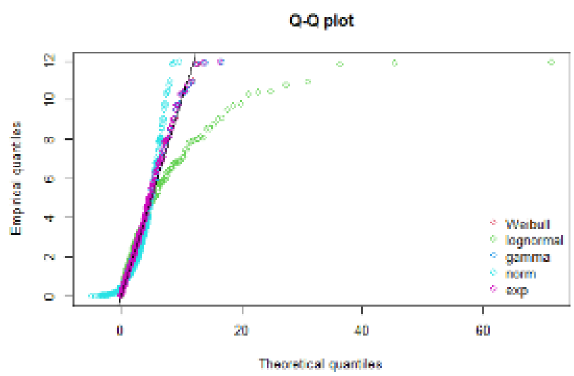
Appendix B:



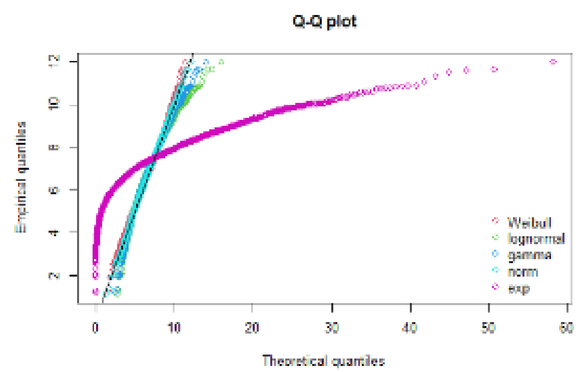
Internal Medicine Crashed



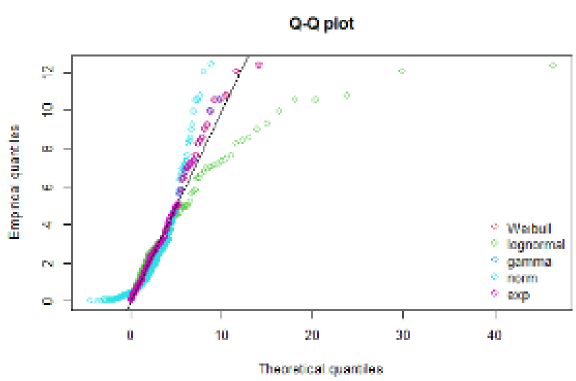
Internal Medicine Not Crashed



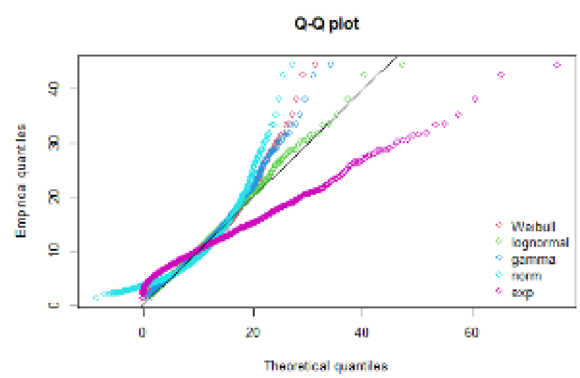
Musculoskeletal Crashed



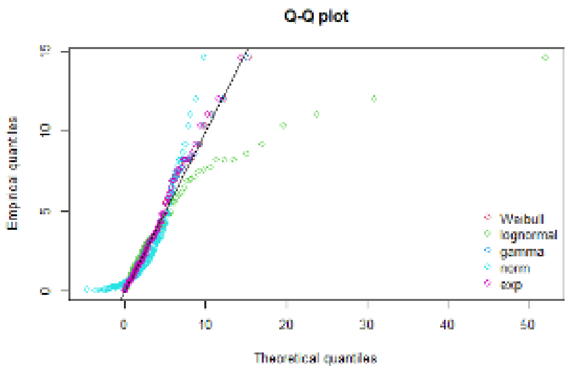
Musculoskeletal Not Crashed



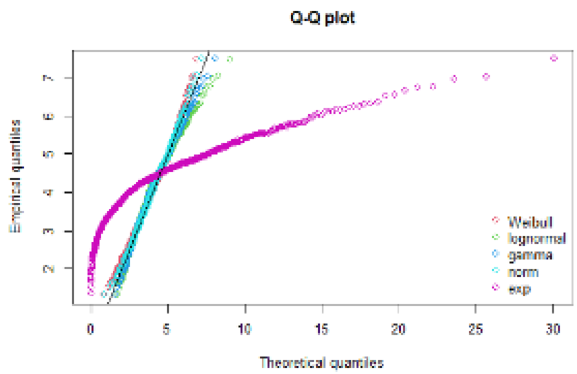
Neurology Crashed



Neurology Not Crashed

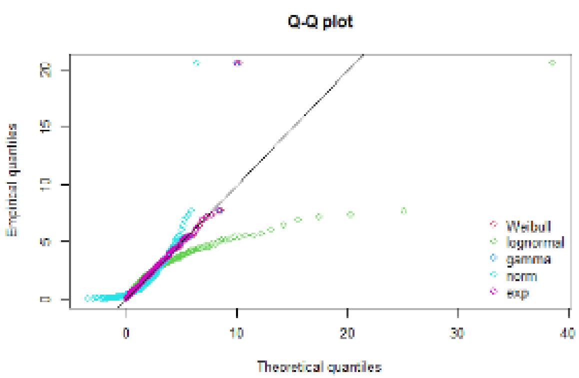


Cardiovascular Crashed

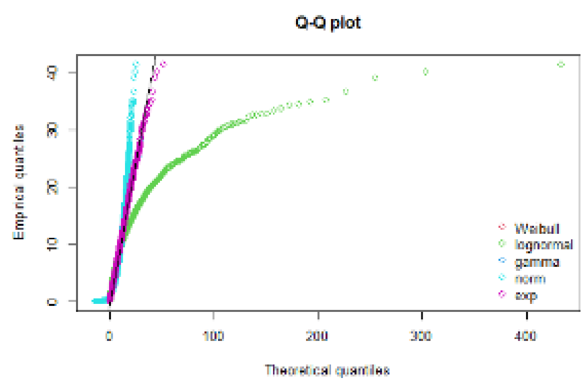


Cardiovascular Not Crashed

Figure 7: QQ plots for all acute data scenarios



ICU Times Patient Died



ICU Times Patient Not Died

Figure 8: QQ plots for service times in ICU

Appendix C:

Table 1. Selected distributions and their parameters for the acute wards

Scenario	Distribution	Parameters
Musculoskeletal0	Normal	Mean = 6.825815 Sd = 1.533714
Musculoskeletal1	Gamma	Alpha = 0.9692772 Beta = 1/0.4021142
Cardiovascular0	Normal	Mean = 4.046612 Sd = 0.970162
Cardiovascular1	Weibull	Shape = 0.9474442 Scale = 2.4623137
Neurology0	Lognormal	Mean = 2.096512 Sd = 0.512613
Neurology1	Weibull	Shape = 0.987769 Scale = 2.184903
InternalMedicine0	Weibull	Shape = 0.9951294 Scale = 5.3717486
InternalMedicine1	Gamma	Shape = 0.9772230 Rate = 0.4466063

Table 2. Selected distributions and their parameters for the ICU service times

Scenario	Distribution	Parameters
ICUTimes0	Gamma	Alpha = 0.9991930 Beta = 1/0.1918584
ICUTimes1	Gamma	Alpha = 0.9960113 Beta = 1/0.6881215

Appendix D:

Table 3. Summary of Response Averages and Half-Widths for the Base Model

Statistic	Ward	Average (or Observations)	Half-Width
Average Waiting Time	Internal Medicine	0.110013	0.0285
	Musculoskeletal	0.2007	0.0283
	Neurology	0.469544	0.0967
	Cardiovascular	0.0204198	0.0057
Queue Length	Internal Medicine	0.72583	0.1899
	Musculoskeletal	0.990276	0.1443
	Neurology	1.58408	0.3336
	Cardiovascular	0.0369321	0.0103
Bed Utilization	Internal Medicine	79.7562	1.0219
	Musculoskeletal	84.2056	0.6888
	Neurology	85.9804	0.9008
	Cardiovascular	52.5796	0.8647
	ICU	85.0242	1.0793
Proportion of External Patients Turned Away	—	0.378399	0.0311
Proportion of ICU Patients Leaving the System to AltHospital	—	0.314159	0.0058

Average ICU Beds Blocked (Number of Observations)	—	246.0750	21.1152

Appendix E:

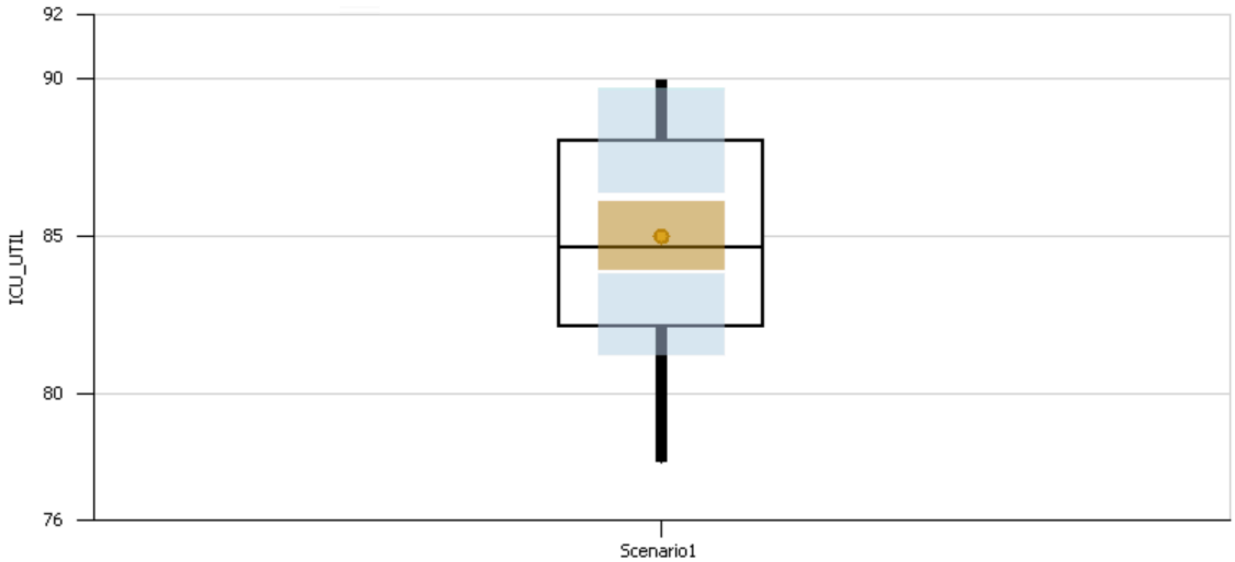


Figure 9: Confidence interval for ICU Util in Base model

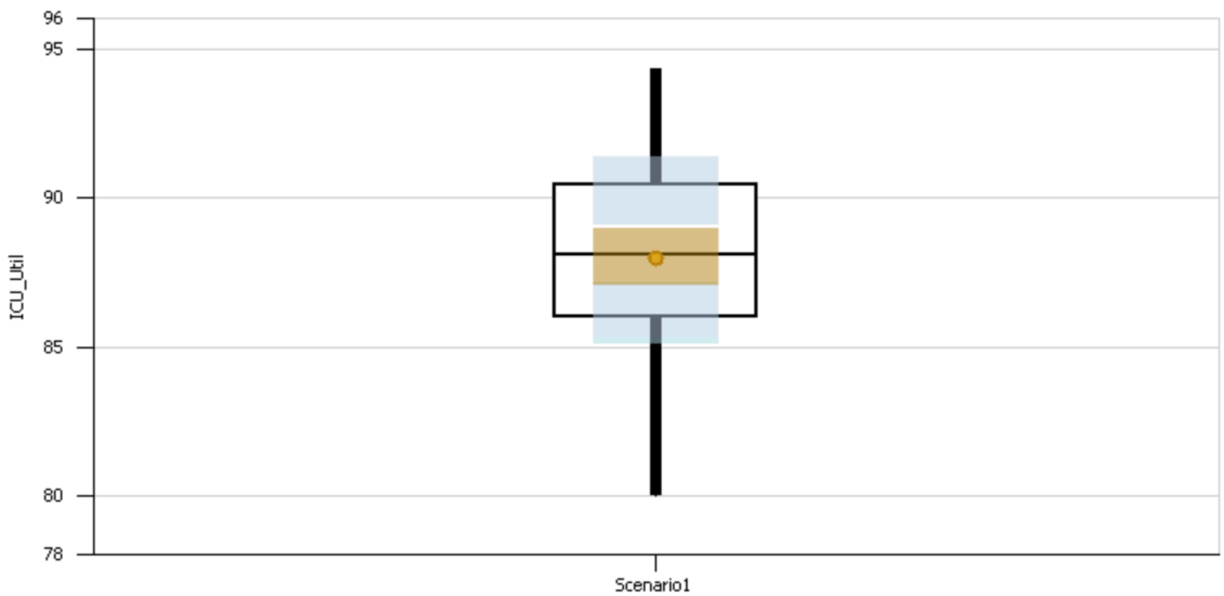


Figure 10: Confidence interval for ICU Util in scenario 1

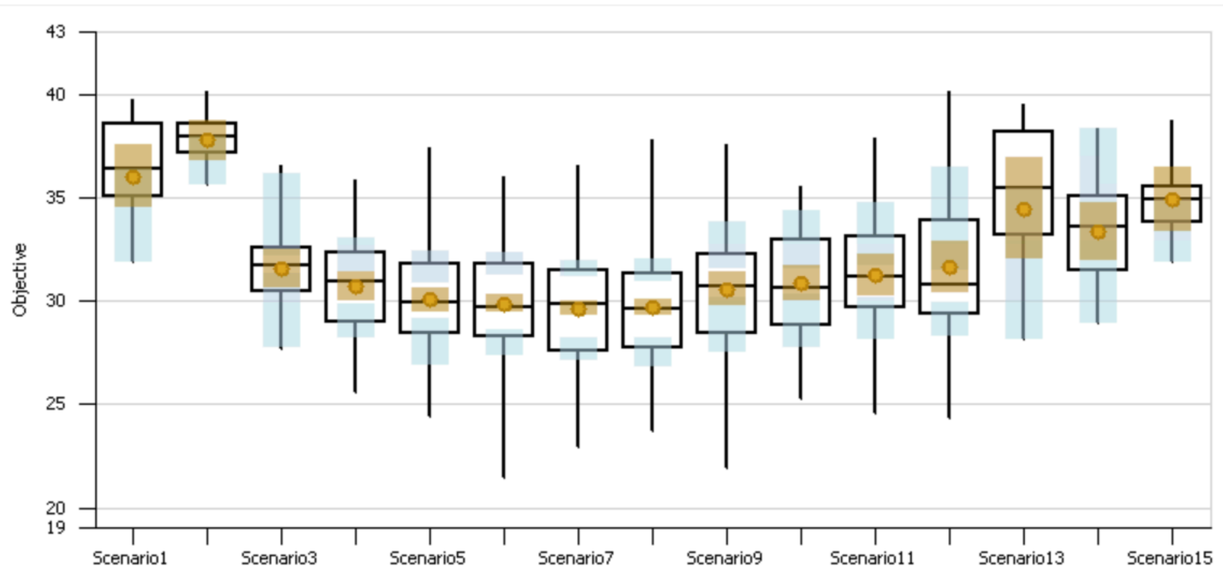


Figure 11: Confidence intervals for the objective function in scenario 2

Appendix F:

Table 4. Comparison of Responses Between Base Model & Scenario 1 with Half-Widths

Responses	**Base Model**	**Scenario 1**	**Scenario 1: Half-Widths**
Average Waiting Time in Acute Ward	0.2001692 (avg)	0.124219	0.0152
Queue Length of Acute Ward	0.834279525 (avg)	2.18354	0.2703
Bed Utilization of Acute Ward	75.63045 (avg)	91.6495	0.3905
Bed Utilization of ICU	85.0242	87.9919	0.9518
Proportion of External Patients Turned Away	0.378399	0.462724	0.0282
Proportion of ICU Patients Leaving the System	0.314159	0.279268	0.0206
Average ICU Beds Blocked (Number of Observations)	246.075	317.925	24.1636

Appendix G:

Table 5. Comparison of different threshold values

Threshold Value (T)	**Objective value**	**Half-Width**	**Threshold Value (T)**	**Objective value**	**Half-Width**
1	36.0535	1.5421	2	37.7965	0.9821
3	31.5641	0.9426	4	30.7124	0.6930
5	30.0877	0.5691	6	29.9012	0.4058
7	29.6708	0.3590	8	29.725	0.3691
9	30.5999	0.8182	10	30.8857	0.8331
11	31.2723	0.9772	12	31.6986	1.2467
13	34.5011	2.4669	14	33.3676	1.3794
15	34.9204	1.5767			