

A Comparison of Queueing Software Packages

By

Chinthanie Fernando Ramasundarahettige

A Major Paper

**Submitted to the Faculty of Graduate and Research
Through the Department of Mathematics and Statistics
In Partial Fulfillment of the Requirements for
The Degree of Master of Science at the
University of Windsor**

Windsor, Ontario, Canada

A Comparison of Queueing Software Packages

By

Chinthanie Fernando Ramasundarahettige

APPROVED BY:

**M. Hlynka, Supervisor
Department of Mathematics and Statistics**

ABSTRACT

In this paper five statistical software packages for queueing theory are compared. The five packages are JPQ, QTP, QTS PLUS, MC- QUEUE and QUICK Q. Installation, documentation, hardware requirement and capabilities of each package are discussed in detail conveniently using some tables. Some results are tested for numerical accuracy and some further findings are presented, so that the user can use the information to select an appropriate system. Finally an overall discussion and conclusions of the research are given.

KEY WORDS: Queueing packages, arrival rate, service rate, waiting time, waiting area, system capacity, birth and death process.

ACKNOWLEDGMENTS

I am indebted to many people who have contributed to my success during my course of study. Special thanks go to my supervisor Dr. M. Hlynka for his guidance, kind support, and encouragement throughout my research study. His direction and assistance made it possible for me to finish this research paper on a reasonable time. Also I wish to thank the departmental reader for his valuable time in reading this paper.

I am grateful to the Department of Mathematics and Statistics at the University of Windsor, for providing financial support in terms of a graduate assistantship throughout my studies. The research assistantship from my supervisor is also appreciated.

My very personal thanks go to my beloved husband and my two sons without whose encouragement and patience, this work would have not been a success. I express my gratitude to my parents and entire family for their love and support during this course of study.

Finally, I wish to give my special thanks to the late Dr. Harold Atkinson for helping me to find the path to this achievement.

TABLE OF CONTENTS

ABSTRACT	iii
ACKNOWLEDGMENTS	iv
LIST OF TABLES AND FIGURES	vi
LIST OF APPENDICES	vii
CHAPTERS	
1. INTRODUCTION	1
2. THEORY	4
3. SOFTWARE INFORMATION	
1. Installation and hardware requirements	9
2. Interface	11
3. Documentation	12
4. Capabilities (Procedures available)	15
4. RESULT TESTING AND FURTHER FINDINGS	36
5. DISCUSSION AND CONCLUSION	42
REFERENCES	45

LIST OF TABLES

1) Input for Single Server Models for QTS PLUS	21
2) Outputs for Single Server Models for QTS PLUS	22
3) Input for Multiple Server Models for QTS PLUS	25
4) Output for Multiple Server Models for QTS PLUS	26
5) Input and Output for Bulk Models for QTS PLUS	28
6) Input and Output for Priority Models for QTS PLUS	29
7) Summary of capabilities	35
8) Output for model $M/M/1$, from the five packages	36
9) Calculated values for model $M/M/1$ using MAPLE	37

LIST OF FIGURES

1) Expected number of customers in the system Vs ρ	39
2) Mean waiting time in the queue Vs ρ	40
3) Mean number of customers in the system Vs ρ for fixed μ	41
4) Mean waiting time in the system Vs ρ for fixed μ	41

LIST OF APPENDICES

(1) a - Notations used for JPQ package.

b - General output for model M/M/1 for JPQ.

c - General output for model M/M/1/K for JPQ.

d - General output for model M/M/C for JPQ .

e - General output for model M/M/inf for JPQ.

f - General output for model M/M/C/K for JPQ.

g - General output for model M/M/C/C for JPQ .

h - General output for model M/M/C/N for JPQ.

(2) a - Probability calculator in QTS Plus.

b - Exponential and Erlanf distribution estimation in QTS PLUS.

c - Output for Markov chain solver in QTS PLUS.

d - Simple Markov queue with state dependent service in QTS PLUS.

e - Erlang input to single Erlang server.

f - Output for bulk models in QTS PLUS.

g - Single server Markovian queue with two priority classes in QTS PLUS.

h - M/M/c queue in series in QTS PLUS.

i - General output for model M/M/C /N/Y for JPQ .

(3) Output for steady state behaviour in MC QUEUE.

(4) a- List of functions in QTP

b – Sample ouput for model M/M/1 in QTP

Chapter 1

Introduction

All of us have experiences of waiting in lines (queues). We wait in lines at banks, at cashier counters, at instant teller machines, in traffic jams, at restaurants. Customers do not like to wait in line and most businesses do not like their customers to have to wait for a long time. Sometimes waiting lines can cause loss of business since customers who see a long queue may decide not to join the queue. There can be more than one reason for long lines. The number of servers might not be large enough, the servers may be slow, the arrival rate may be very high. There are many natural questions to ask and answer. How many servers are needed? What should the waiting area size be? How long can a customer expect to wait in line, to get service? What is the probability that the server is busy? What is the probability that the server is empty? Queueing theory attempts to answer these questions through mathematical analysis.

In this paper, several Queueing software packages (**JPQ**, **QTS Plus**, **QTP**, **Quick Q** and **MC Queue**) are compared on features such as installation, hardware requirements, documentation, available procedures, accuracy, user friendliness, reliability, technical support and capabilities. This comparison could be extremely useful for one who needs to choose a package to answer questions about queueing models.

The book “Fundamentals of Queueing Theory”(second edition,1998) by Donald Gross and Carl M. Harris states that in most cases, six basic characteristics of queueing processes provide an adequate description of a queueing system. These are as follows.

- 1) Arrival pattern – We should know the interarrival times and whether batch (or bulk) arrivals are possible, if jockeying for position is possible, if the arrival pattern changes with time (stationary or non stationary).
- 2) Service pattern – We should know the service time distribution and whether the service can be provided in batch or not and whether the service rate depends on the number of customers in the queue.

- 3) Queue discipline – We should know the manner in which the customers are selected for service when they are in the queue (first come first serve, last come first serve or priority based).
- 4) System capacity – We should know the size of the waiting room.
- 5) Number of service channels or servers
- 6) Number of service stages – we should know whether the queueing system has single stage service or multi stage service.

1.1 Notation : (taken from “Introduction to Probability Models”, 8th edition, by Sheldon M. Ross).

1.1.1 Interarrival time

T_1 - time of the first event

T_n - elapsed time between the $(n-1)^{st}$ and n^{th} event.

1.1.2 Waiting time

Waiting time is the time from the arrival of a customer until it enters service.

1.1.3 Time of the n^{th} event - S_n

$$S_n = \sum_{i=1}^n T_i$$

1.1.4 Kendall notation (1953) for describing a queueing process:

A/B/X/Y/Z , where A = arrival process

B = Service process

X = number of servers

Y = size of the waiting area

Z=population size from which customers come

1.1.5 λ = Average arrival rate of entering customers.

μ = Average service rate of a server

Utility factor/Traffic intensity (ρ) - (for M/M/c Model)

$$\rho = \frac{\lambda}{c\mu}, \text{ where } c = \text{number of servers}$$

E(L) = Average number of customers in the system (including the customers in service)

E(L_q)= Average number of customers waiting in the queue (excluding customers in service)

E(W) = Average amount of time that customers spend in the system

E(W_q)= Average amount of time that customers spend waiting in the queue.

1.1.6 X(t) denotes the number of customers in the system at time “t” and,

$P_n = \lim_{t \rightarrow \infty} P\{X(t) = n\}$, where “ P_n ” denotes the limiting probability that there will be exactly “n” customers in the system (steady state probability).

1.2 Organization of Major Paper

This major paper consists of five chapters. In Chapter 1, a brief discussion of queueing models and the objective of the paper is given. In Chapter 2, some of the theory used for the five packages discussed in this paper is presented. Chapter 3 consists of information about the software of the five packages discussed, including installation, interface, documentation, and available procedures or capabilities. Chapter 4 consists of result testing and further findings which are useful for a user who may wish to modify the existing system. The overall discussion and conclusion are presented in Chapter 5.

Chapter 2

THEORY

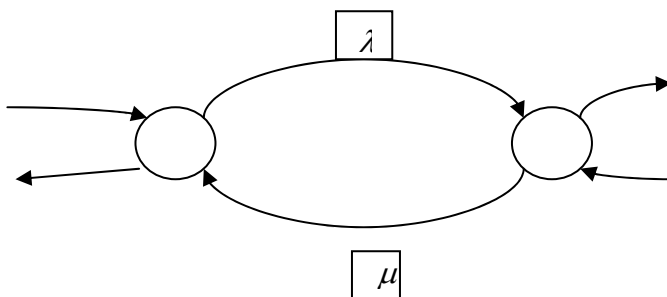
In most queueing models, the stationary equations are obtained using a flow balance procedure. The analysis looks at a given state and requires that average total flow into the state be equal to average flow out of the state if steady-state conditions exist. Using this concept we can find the steady state probability for a given queueing model and system performance measures such as L , L_q , W , W_q . Steady state probabilities and system performance measures for the M/M/1 queueing model are calculated below and for the other models M/M/c, M/M/ ∞ , M/M/1/c, M/M/c/c only expressions for these measures are given (taken from the book “Fundamentals of Queueing Theory” by Gross and Harris).

2.1 M/M/1

1st M – interarrival process is Markovian (exponential)

2nd M – service distribution is exponential

and 1 indicates a **Single** server



Rate matrix = Q

$$\mathbf{Q} = \begin{matrix} & \mathbf{0} & \mathbf{1} & \mathbf{2} & \mathbf{3} & & \\ \begin{bmatrix} -\lambda & \lambda & 0 & 0 & \Lambda & 0 \\ \mu & -(\lambda + \mu) & \lambda & 0 & \Lambda & 0 \\ 0 & \mu & -(\lambda + \mu) & \lambda & \Lambda & 0 \\ 0 & 0 & \mu & -(\lambda + \mu) & \Lambda & 0 \\ 0 & 0 & 0 & \mu & \Lambda & \Lambda \\ \mathbf{M} & \mathbf{M} & \mathbf{M} & \mathbf{M} & \mathbf{M} & \mathbf{M} \end{bmatrix} & & & & & & \end{matrix}$$

Balance equation between state **(n-1)** and state **n**;

$$p_{n-1}\lambda = p_n\mu$$

$$p_n = (\lambda / \mu)p_{n-1}$$

M

$$p_i = (\lambda / \mu)^i p_0$$

$$\rho = \lambda / \mu.$$

$$\Theta \sum_{i=0}^{\infty} p_i = 1$$

$$\Rightarrow p_0 \sum_{i=0}^{\infty} \rho^i = 1$$

$$\Rightarrow p_0 = (1 - \rho)$$

$$p_n = \rho^n (1 - \rho).$$

Expected wait for a customer arriving at random to complete service = E (W).

Expected length of the queueing system = E (L).

$$E(W) = \sum_{i=0}^{\infty} P_i E(W / i \text{ other customers})$$

$$\begin{aligned}
 E(W) &= \sum_{i=0}^{\infty} p_i \frac{(i+1)}{\mu} \\
 &= \frac{1}{\mu} \sum_{i=0}^{\infty} i p_i + \frac{1}{\mu} \\
 &= \frac{1}{\mu} (E(L) + 1)
 \end{aligned}$$

Using the Little's formula,

$$E(L) = \lambda E(W)$$

And hence solving for E(L) yields,

$$\begin{aligned} E(L) &= \lambda E(W) \\ &= \frac{\lambda}{\mu}(E(L)+1) \end{aligned}$$

Expected length of queue (waiting only) = E(L_q)

$$E(L_q) = \sum_{i=1}^{\infty} p_i (i-1)$$

$$E(L_q) = \frac{\rho^2}{(1-\rho)}$$

Expected wait for a randomly arriving customer to enter service = E(W_q)

$$E(W_q) = \sum_{i=0}^{\infty} p_i E(W / i \text{ other customers})$$

$$E(W_q) = \frac{\lambda}{\mu(\mu - \lambda)}$$

2.2 M/M/c , c = Number of servers

$$p_0 = \left\{ \sum_{i=0}^{c-1} \frac{(c\rho)^i}{i!} + \frac{(c\rho)^c}{c!(1-\rho)} \right\}^{-1}$$

$$L_q = \left\{ \frac{(c\rho)^c \rho}{c!(1-\rho)^2} \right\} p_0$$

$$W_q = L_q / \lambda$$

$$W = W_q + 1/\mu$$

$$L = L_q + \lambda/\mu$$

2.3 M/M/∞ , Number of servers = ∞

$$p_0 = \left(\sum_{n=0}^{\infty} r^n / n! \right)^{-1}, \text{ where } r = \lambda / \mu$$

$$p_n = (r^n e^{-r}) / n!$$

$$L = \lambda / \mu$$

$$W = 1 / \mu$$

$$L_q = 0 = W_q$$

2.4 M/M/c/K

$$p_0 = \left\{ \left(\sum_{n=0}^{c-1} \lambda^n / n! \mu^n \right) + \left(\sum_{n=c}^K \lambda^n / c! \mu^n c^{n-c} \right) \right\}^{-1}$$

$$p_n = \begin{cases} \lambda^n p_0 / n! \mu^n, & 1 \leq n < c \\ (\lambda^n / c^{n-c} \mu^n c!) p_0, & c \leq n \leq K \end{cases}$$

2.5 State dependent service

Here, the mean service rate depends on the number of customers in the queue. Suppose μ_n is given as follows.

$$\mu_n = \mu_1, \quad 1 \leq n < k,$$

$$\mu, \quad n \geq k$$

Assume a single server. Let $\rho = \frac{\lambda}{\mu}$, $\rho_1 = \frac{\lambda}{\mu_1}$. Then,

$$P_0 = \begin{cases} \left\{ \left[\frac{(1-\rho_1^K)}{(1-\rho_1)} \right] + \left[\frac{(\rho_1 \rho_1^{K-1})}{(1-\rho)} \right] \right\}^{-1} & (\rho_1 \neq 1, \rho < 1) \\ [K + (\rho/1-\rho)]^{-1} & (\rho_1 = 1, \rho < 1) \end{cases}$$

Where both $\rho < 1$ and $\rho_1 < 1$,

$$L = P_0 \left\{ \left(\sum_{n=0}^{k-1} n \rho_1^n \right) + \left(\sum_{n=k}^{\infty} n \rho_1^{k-1} \rho^{n-k-1} \right) \right\},$$

$$L_q = L - (1 - P_0),$$

$$W_q = L_q / \lambda,$$

$$W = W_q + (1 - P_0) / \lambda.$$

3.1 Erlang probability distribution function

The Erlang distribution is the distribution of the sum of k independent identically distributed random variables each having an exponential distribution.

The probability density function of the Erlang distribution is

$$f(x; k, \lambda) = \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!} \quad \text{for } x > 0.$$

3.2 Hyperexponential probability distribution function

A random variable X is hyperexponentially distributed if X is with probability p_i , $i=1, 2, \dots, k$ an exponential random variable X_i with mean $1/\mu_i$.

The probability density function of the Hyperexponential distribution is

$$f(x) = \sum_{i=1}^k p_i \mu_i e^{-\mu_i x}, \quad \text{for } x > 0$$

Chapter 3

Software Information

3.1 INSTALLATION AND HARDWARE REQUIREMENT

3.1.1 JPQ

Java Powered Queueing, was developed by Muhammad El –Taha and Bacel Maddah of University of Southern Maine .This is still under development as of the date 4/06/05 and it is in version 1.0 (Beta) .We do not need to download or install the software to run this package. When we double click the site <http://www.usm.maine.edu/math/JPQ/about.html> it automatically goes to the page where we can see the java applet. User can do calculations necessary in this page. But in my case I did not need to download this to my computer to operate this package but I can not operate JPQ in some other coputers.

3.1.2 QTS PLUS

This was based on work by Donald Gross and Carl M. Harris of George Mason University. The user can find the down-loading instructions which are given in the site <http://www.geocities.com/qtsplus> . This software is in two versions and user the has to decide which version of the software is applicable depending on the user's spreadsheet version. QTS Plus is in version 2.1, contains bug fixes and new models and run successfully in Excel 97 or above, needs 7.5 MB disc space, and requires additional Excel Add-Ins : Analysis Toolkit-VBA.

This software is located at John Wiley and Sons FTP server. To download the software, click on the desired version. The downloaded file can be placed anywhere in the local hard drive. To install the software, double-click the downloaded file (this is a self-extracting zip file) using the Windows File Manager or Explorer. Either accept or change the default given in the popup window and press the “Un Zip” button to complete the installation.

3.1.3 MCQueue

This is educational software for Markov Chains and Queues developed by Henk Tijms of Vrije University, The Netherlands. Downloading instructions are given in the site <http://staff.feweb.vu.nl/tijms/> . Download the zip file by double clicking it and place it anywhere in the local drive. It allocates 654KB of the disc space. We have to extract the zip file whenever we want to use the package. After extracting the zip file , double click the “**OR Stat**” to run the software.

3.1.4 Quick-Q

Quick-Q was developed by Brian Fox . It can be download from the “Statlib” site <http://lib.stat.cmu.edu/dos/general/> . This software needs Windows 3.11 or Windows 95 or later and needs 1.2 MB disc space. This is also a zip file and can be placed anywhere in the hard drive. But here too, we need to extract the zip file whenever we want to use the package. Double click to extract the zip file and double click “Q” to run the package.

3.1.5 QTP-Queueing ToolPak

QTP was developed by Armann Ingolfsson and Fraser Gallop of University of Alberta. Download instructions are given in the site <http://www.bus.ualberta.ca/aigolfsson/QTP>, and it allocates 650 KB disc space. It successfully work in Windows 2000 and Windows XP ,with Excel 2000 and Excel XP. Getting QTP 4.0 add-in to work on the computer involves two steps as downloading the **add-in** files and loading the **add-in** in Excel. Double click the **qtp.exe** in the given site, place it in a desired folder, and unzip the file. To Load QTP 4.0, open Excel and choose “add-ins” from the “tools” menu. Browse the unzipped file and select qtp.xll and click “ok”. “QTP” will appear in the menu and by that we can verify Queueing ToolPak has been loaded correctly.

3.2 INTERFACE

3.2.1 JPQ

This is a JAVA applet and does not have a menu. In the JAVA applet, the user should select the required queueing model and then enter the required inputs. Here input and output can be identified easily since the screen is divided into two parts - INPUT and RESULT.

3.2.2 QTS Plus

This has Windows-like interface. The user can only use the mouse to maneuver within this window which contains pull down menus. When the user selects the queueing model, the package goes to Excel environment and from there the user can input values and get the outputs. It uses different colors for output and input, so the user can easily differentiate the output from the input.

3.2.3 MC QUEUE

MC Queue has a Windows-like interface and cascading pull-down menus. There is no need to type any command to execute the procedures. Results will appear in a different window.

3.2.4 QUICK Q

QUICK Q also has a Window-like interface and contains pull down menus. The animation part in the pop-up window gives the user, a kind of a feeling that he/she really is in a queue.

3.2.5 QTP

This really is an Excel work sheet where the user has to type every command that is needed. There are no pull down menus. The user can arrange the work sheet according to his/her requirements. This is not as user-friendly as the above discussed packages but the user has option to work on his own.

3.3 DOCUMENTATION

3.3.1 JPQ

This does not contain either a User's Guide or a help menu. It only contains a description of the notation used within the package. When interpreting results, it would be helpful if statistical theory used by the package were given. This package is not valuable for a person who does not have much knowledge about queueing theory since interpretation is not easy.

3.3.2 QTS Plus

For this software documentation consists of a Help menu, a Tutor and an Index. The help menu consists of an introduction, disclaimer, dedication, new or change models section (comparing with the old version) , run-time configuration parameters, model notes (a little description about the restrictions on the models) , simulation models (description about the simulation models G/G/1, G/G/C, single node simulation model and user defined distribution functions for the G/G/1 and G/G/C), a description of the QTS tutor, other information (programming conventions, problem solving advice, and how to contact the author). The "Tutor" demonstrates how queueing models are applied to real world problems. It consists of examples such as M/M/1, M/M/c, M/D/1... etc. Some of the illustrations are from examples found in "Fundamentals in queueing theory" 3rd ed., by Gross and Harris. The "Index" consists of a complete alphabetical listing of all models in QTS Plus. All the documentation is well written and easy to understand. All examples in the tutor have satisfactory explanation and give the first time user a good understanding about how to use the package. The on line help available is very useful for the user. Problem solving advice is also very useful for the user to take correct approach to solve problems encountered. However, the documentation does not contain technical details describing the statistical formulas used in the calculations.

3.3.3 MC Queue

MC Queue documentation contains only a help menu. This help menu describes Markov chains, Queues and modules in MC Queue in detail. The Markov chain section contains an introduction, section on chain structure, steady state behavior, transient state behavior, first passage time probability and references. The queueing section contains a detailed description of the M/G/1 Queue, D/G/1 Queue, M/M/C Queue, D/M/C Queue, G/M/C Queue, finite capacity queues, finite source queues, transient M/M/1 queues, M/M/1 Queues, M/M/C/C+N Graphs, INPUT parameters, performance measures, and references. The descriptions are very useful for the user to understand and interpret the results. This package, also has an on-line help capability. If this package contained some examples like QTS Plus, it would be more helpful for a first time user. The package does not have trouble shooting advice as in QTS Plus.

3.3.4 QUICK Q

The only documentation available is the on line help and it works as a user guide. This help menu consists of “**contents**”, “**search**”, “**about Quick Q**”. Under the contents we find (1)an “**overview**” which describes Quick Q and gives information about the author, detailed instructions of how to use the Quick Q (“**using Quick Q**”) (including a detailed example) and (2) a “**Queuing theory**” part, which describes the types of the queues (in detail with a cascading menu where we can find all the formulas), how the solutions appear in the screen and descriptions of the formulas, variables, distributions used in this package, planned enhancement where we can find a description of the Quick Q versions available and a bibliography which gives the names of the books used to make this package. All the instructions and descriptions are easy to understand and the detailed examples gives the first time user very good guidance on how to use this package. This help menu gives almost every detail that a user should know to operate this package. If this package had trouble shooting advice, like in QTS Plus, it would be useful.

3.3.5 QTP

Here too the on line help menu acts as a user guide. The help menu consists of “Queueing Tool Pak help”, “functions reference” and “error messages”. In the “Queueing Tool Pak help” we can find a list of functions (QTPMMS, QTPMMS_util , QTPMMS_L...etc), the function wizard which describes functions in this package, a deacription of all the arguments used, assumptions of the two categories QTPMMS and QTRGGS, two tutorials which contain detailed instructions for two questions, a glossary which lists some common terms used, and credits for the persons who did this package.

In the “function reference”, the user can find list of functions available in this package with a cascading menu. The user can find almost all the details in Queueing theory that are needed to use this package. As in QUICK Q, this package does not provide the statistical formulas used for the procedures. If it contained this information it would be very helpful for the user to describe the output statistically.

In the “error message” part the user can find a detailed description of the error messages which appear in this package in alphabetical order. This is very good because then the user can identify mistakes.

The two tutorials in the help section are very useful to a first time user since they give all the details needed in very simple instructions that are easy to understand. Through these two examples the user can get a better knowledge about how to use the procedures in this package.

Comparing the documentation part of all these packages, QTS Plus, QUICK Q and QTP give the user a comprehensive knowledge about the package while MC Queue gives less information and JPQ does not give much information to the user.

3.4 CAPABILITIES(PROCEDURES AVAILABLE)

3.4.1 JPQ

As mentioned earlier, this is still under development and it facilitates the following queueing models, M/M/1, M/M/1/K, M/M/C, M/M/inf, M/M/C/K, M/M/C/C, M/M/C/N, M/M/C/N/Y. First, the user needs to select the required model from the Java applet. Then the user needs to type the input values in the relevant boxes. Then press the “compute” button to get the output. The user can plot “ P_n Vs n ” for every queueing model using the “Plot P_n ” button in the input output page and the graph will appear in a separate Java applet. Notation used by this package appear in Appendix (1) - a .

3.4.1.1 In the M/M/1 (Single –Server, Infinite waiting room, Markovian Queue) model the user needs to input Mean arrival rate (λ), Mean service rate (μ), number of customers in the system (n), waiting time in the system (t), waiting time in the queue (t_q). But “ n ”, “ t ” and “ t_q ” are not defined in the notation and it seems that these do not affect the other calculations (L, L_q, W, W_q). If the user needs to find the probability of number of customers in the system at a random time greater than “ n ”, $\{P(X > n)\}$, the user needs to input that “ n ” as the value of “ n ” in M/M/1. This “ n ” should be an integer and λ, μ, t and t_q can be any number in decimal form. But all these values are required as input to get an answer, whether the user needs the probability or not. Otherwise, the user will get an error message that some of the required data are missing.

The outputs of the M/M/1 model gives values of the traffic intensity (ρ), expected system size (L_q), standard deviation of queue size at random time (σ_{X_q})(where X_q indicates the queue size in steady state), expected waiting time in the queue (W_q), standard deviation of the waiting time in queue of a random customer (σ_{T_q}) where T_q indicates the expected waiting time in queue of a random customer, expected system size (L_q), standard deviation of the number of customers in the system in steady state (σ_X) where X indicates the number of customers in the system in steady state, expected waiting time in the queue (W), standard deviation of the waiting time in the system (σ_T)

where T indicates waiting time in the system of a random customer, probability of number of customers in the system at a random time is greater than the given “ n ”(input value) $\{P(X>n)\}$, probability of waiting time in the system of a random customer is greater than the given “ t ” $\{P(T>t)\}$, probability of the waiting time in queue of a random customer is greater than the given “ t_q ” $\{P(T_q > t_q)\}$.

To get values for the above quantities, the condition $\lambda < \mu$ should be satisfied. If not, the package will give an error message that “ $\text{Rho} > 1$ and cannot converge to a solution”. When “ n ” is not an integer it gives an error message that “Format is not correct”. I found that when “ $n > 173$ ” it consider “ n ” to be infinity. The general outputs for some different input values are given in Appendix (1)- b .

3.4.1.2 In the M/M/1/K model (Single-Server, Finite waiting room, Markovian Queue) the input values are λ, μ, K and this K should be an integer (please refer the Appendix (1)-a for the notation.). The output values for $\rho, P_K, \lambda_e, L_q, W_q, L, W$. Here there is no restriction on the input values λ and μ which be written in decimal form. Output for different values of λ, μ are given in Appendix (1)-c.

3.4.1.3 In the M/M/c model (Multiple-Server, Infinite waiting room, Markovian Queue), the user has to input values for $\lambda, \mu, c, n, t, t_q$. As the output ,the user will values for $\rho, L_q, \sigma_{X_q}, W_q, \sigma_{T_q}, L, \sigma_X, W, \sigma_T,$

$P(X \geq n), P(T > t), P(T_q > t_q)$. In this case “ c and n ” should be integers and $\lambda < c\mu$ in order to get an answer. Otherwise the user will get an error message . The general output for different values of λ, μ is given in Appendix (1)-d.

3.4.1.4 In the M/M/inf model (Infinite –Server, Markovian Queue), the user needs to input the values of λ and μ ,and values of L_q, W_q, L, W will returned as outputs. Some general outputs are in Appendix (1)–e.

3.4.1.5 In an M/M/c/K model (Multiple-Server, Finite waiting room, Markovian Queue), the user needs to input values of λ, μ, c, k and will get the values of $\rho, P_K, \lambda_e, L_q, W$ and L as the outputs. In this model also “ c ” and “ k ” should be integers and the condition $\lambda < c\mu$ should be satisfied to get an answer. Otherwise the user will get error messages. General output for different values of λ, μ are given in Appendix (1)-f.

3.4.1.6 In the M/M/c/c model, the user needs to input the values of λ, μ, c and will get values for $\rho, P_C, \lambda_e, L_q, W_q, L, W$ as the output. Here the only restriction is that “c” should be an integer. Some general outputs are in Appendix (1)-g.

3.4.1.7 For the M/M/c/N model (Machine interference model), values of λ, μ, c, N should be entered as input values to get values for $\lambda_e, L_q, W_q, L, W$ as the output. For this model “c” and “N” should be integers and the condition $c < N$ should be satisfied in order to get an answer. Otherwise the user will get an error message (if $c > N$) “Invalid Format or Number, Please Reenter, $c > N$, Not feasible”. Some general outputs are in Appendix (1)-h.

3.4.1.8 For the M/M/c/N/Y (Machines Interference Model with spare machines) model, values of λ, μ, c, N, Y are needed as input values, and values for λ_e, L_q, W_q, L and W are the output values. For this model also “c, Y and N” should be integers. General output for different values of λ, μ, c and N are given in the Appendix (1)-i.

3.4.2 QTS PLUS

To use this package, the user needs to select the “QTS Plus” Excel sheet from the extracted files, enable the macros first and follow the next instruction given (“press to continue”). Then the user can see the main page of QTS Plus. In that page, there are different models. The user can select a model according to his/her requirements and global configuration parameters (tolerance limit for convergence, upper limit on number of iterations for convergence test, background colors for input and output, switch to protect work sheet and charts) are given.

There are more than 60 models in this package and all of these can be categorized under 7 main categories such as Basic probability models, Single server model, Multiple server models, Bulk models, Priority models, network models and Simulation models. The user needs to select QTS from the menu bar of the Excel sheet and then select the appropriate model from the cascading menu.

3.4.2.1 The Basic probability models section is subdivided again into three parts, Probability calculators, Statistical tools and Markov chain solver.

3.4.2.2 The Poisson calculator, Erlang calculators and Mixed Exponential calculator in QTS PLUS fall under Probability calculator. In the Poisson calculator, the user needs to input the “mean of distribution - λ ” and the plot parameter value. As the output, the user will see the values of “ n , p_n , P_n ” in the same worksheet and user can toggle between the PMF –Poisson probability mass function, and CDF – Poisson cumulative probability charts (here “ p_n ” is the steady state probability of “ n ” customers in the system and “ P_n ” is the cumulative of “ p_n ”). In the Erlang calculator the user needs to input the values of the Erlang scale parameter (β), the Erlang shape parameter (κ) and the plot parameter. The distribution mean, Distribution variance, and coefficient of variation will appear in the same work sheet as the outputs. The user can toggle between the density chart, distribution chart and the plot data. In the Mixed Exponential calculator the user needs to input values of p_i , \mathcal{G}_i and the plot parameter. But here the user needs to press the “solve” button to get the mean value, second moment about zero, variance, and coefficient of variation as the output. Here too the user can toggle between worksheet, Mixed Exponential density chart, Mixed Exponential distribution chart and the plot data.

In the “Statistical tool” part, the user can test a data sample to check whether it comes from an Exponential distribution or from an Erlang distribution with specified parameters. Here, the user needs to input two data sets of approximately equal length and press the “solve” button to get the output. In the output user will get values of the test statistics, and the mean of both Exponential and Erlang distributions and the decision of the statistical test with significant levels $\alpha = .1, .05, .01$. (here H_0 :exponential and H_1 :Erlang parameter 2). Both “Probability calculator” and “Statistical tools” are special features of QTS Plus. See Appendix (2) -a and Appendix (2)-b for sample output given in the package for these two facilities.

The Markov Chain solver in QTS PLUS is capable of handling discrete time Markov chains (including transient analysis), continuous time Markov chain, finite birth-death model, infinite birth-death model and delay analysis for a sample single server queue. In “Discrete time Markov chain”, the user needs to input the number of states and the transition probability matrix and press “solve” button to get the output. The output gives the steady state probabilities in the same worksheet. In the case of “Transient analysis”, the user needs to input the number of transitions in addition to the above two inputs. In

this case, the output will appear in a separate worksheet and user can toggle between these two pages. In “continuous time Markov chain”, the user needs to input the number of states and the rate matrix “Q”. The user will get the steady state probability as the output. In “Finite birth-death model”, the user needs to input number of transitions, birth and death rates, and to obtain the state probabilities and the probability chart, needs to press the “solve” button. In “Infinite birth-death model”, the user needs to input the birth/death rates, highest state probability wanted, birth and death rates. As output, the user will get the state probabilities and the infinite probability histogram. Sample output given in the package is in Appendix(2) -c.

3.4.2.3 Under “Single sever models” QTS PLUS is capable of operating nineteen different models such as M/M/1 (calculates major measures of effectiveness and distributions for system size and wait time), Transient M/M/1 (conducts a transient analysis for M/M/1), M/M/1/K (calculates major measures of effectiveness and distributions for system size and wait time), M/M/1 with Balking, Markov single server Finite source Queue without spare parts and with spare parts, Simple Markov Queue with State dependent service, General State Dependent birth and death queue, M/G/1, M/G/1: Sensitivity analysis(determining the effective of service time variation on system performance measures for standard M/G/1 queue using Pollaczek-Khinchin formula), M/D/1, M/E(k)/1, M/H/1, D/M/1, E(j)/M/1 , H/M/1(Poisson input to single server queue with Hyperexponential service times, unlimited capacity) , G//M/1/, E(j)/E(k)/1 and G/G/1:Approximation model. The values which the user needs to input to different models and the outputs of the package are given in Table1 and in Table2. The following acronyms define the names of the different models in those tables.

Acronyms used for Table 1 and Table2.

A	M/M/1
B	M/M/1 Transient
C	M/M/1/K
D	M/M/1 with balking
E	Markov single server,Finite source queue -without spares
F	Markov single server, Finite source queue -with spares
G	Simple Markov queue with State dependent service
H	General State dependent Birth-Death queue
I	M/G/1
J	M/G/1 -Sensitivity analysis
K	M/D/1
L	M/E(k)/1
M	M/H/1
N	D/M/1
O	E(j)/M/1
P	H/M/1 -Balanced
Q	H/M/1- Mixed
R	G/M/1
S	E(j)/E(k)/1
T	G/G/1

Table 1:Input for Single Server Models for QTS PLUS

INPUT	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
Arrival rate (λ)	V	V	V	V	V	..	V	V	V	V	V	V	V	V
Mean service time ($1/\mu$)	V	V	V	V	V	V	V	..	V	V	V	V	V	V	V	V	V	V
Time reference point (t)	..	V
Initial system state (i_0)	..	V
Target state for the transient probability (n)	..	V
maximum capacity of the system($K>1$)	V
Balking function (select appropriate one)	V
Balking parameter (a)	V
Mean interarrival time for a single customer ($1/\lambda$)	V	V	V
Maximum # of customers in the system (M)	V	V
Specific time for delay distribution calculation (t)	V	V
Number of spares (Y)	V
First mean time to complete service ($1/\mu_1$)	V
Second mean time to complete service ($1/\mu_2$)	V
Switch number of customers	V
Number of Birth/Death Rates:	V
Highest State Probability Wanted:	V
Service time variance (σ^2)	V
Erlang shape parameter (k)	V	V	V
Erlang service parameter	V
Variance of interarrival time	V
Variance of service time	V
Test time	V
t, a (t) (interarrival time,probability)	V
Proportion of Type I customers (q)	V
Number of alternate phases (n)	V
Number of intearrival probabilitis****	V
Maximum size for probability chart	V	V	V	V	V	V	V	V	V	V	V	V	V	..
Total time horizon for probability plotting	V	..	V	V	..	V	V	V	V	V	V	V	..

Table 2:Outputs for Single Server Models for QTS PLUS

OUTPUTS	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
Mean interarrival time ($1/\lambda$)	V	V	V	V	V	..	V	V	V	V	V	V	V	V	V	..
Service rate (μ)	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
Server utilization (ρ)	V	V	V	V	..	V	V	V	V	V
Mean number of customers in the system (L)	V	..	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
Mean number of customers in the queue (Lq)	V	..	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
Mean waiting time (W)	V	..	V	V	V	V	V	V	..	V	V	V	V	V	V	V	V	V	V	V
Mean waiting time in the queue (Wq)	V	..	V	V	V	V	V	V	..	V	V	V	V	V	V	V	V	V	V	V
Fraction of time the server is idle (ρ_0)	V	V	V	V	V	..	V	V	V	V	V	V	V	V	V	V	V	V
Expected non-empty queue size (Lq')	V
Mean length of busy period (B)	V	V	V	..	V	V
$p_n(t)$ = desired transient probability	..	V
p_n = limiting value of state probability	..	V
Effective arrival rate (λ_{eff})	V	V	V	V	..	V
Traffic intensity (ρ) [NEED NOT BE < 1]	..	V
Probability that the system is full (ρ_K)	V	V
Expected number turned away/unit time	V
Probability that an arrival's line delay exceeds t	V	V
Mean arrival rate (λ)	V	V	V	V	V	..	V
First mean service rate (μ_1)	V	V	V	V	V	V	..	V
Second mean service rate (μ_2)	V
Overall mean service rate (μ_{eff})	V
First traffic intensity (ρ_1)	V
Second traffic intensity ($\rho_2 < 1$)	V
system utilization	V
Service time Coefficient of Variation ($\sigma*\mu$)	V
Coefficient of variation of the service distribution	V
Variance of the system unit (Var_W)	V
Variance of the line delay (Var_W)	V
Roots of generating function denominator	V	V	V	V	V	V	V
Fraction of time the server is busy [MUST BE < 1](ρ)	V	V	V	V	V
Expected systems size seen by an arrival ($L^{(A)}$)	V	V	V	V
Expected queue size seen by an arrival ($Lq^{(A)}$)	V	V	V	V
Expected non-empty queue size (L'q)	V	V	V
Probability of no wait (q_0)	V	..
Probability that delay > t (PTWq)	V	..

Table 2- contd.

OUTPUTS	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
charts																				
Size distribution chart	v	..	v	v	v	v	v	v	v	v	..	v	v	..	v	..	v	..
Time distribution chart	v	..	v	v
Plot data	v	..	v	v	v	..	v	v	..	v	..	v
Probability chart	v
W chart	v	v	v	v	..	v
W(t) chart	v	v	v	v	v	..	v	..

In Table 1, *** indicates, for G/M/1 model when the user input the value for number of inerarrival probabilities(n), the package allows the user to input the values for inerarrival points and interarrival probabilities equal to that “n”.In all cases, input values can be any number, real or fractional (it converts it to a real number by itself) and QTS PLUS assumes that every queue has a FIFO (first-in first-out) queueing discipline.The handling “Simple Markov queue with state dependent service” (where service rate depends on the state of the system) and “Erlangian models” ($M/E_k/1, E_k/M/1, E_j/E_k/1$) are other special abilities of this package. Sample outputs for these two models are given in Appendix (2) -d and (2) - e.

3.4.2.4 Under “Multiple- server Model” QTS PLUS has 17 models as described below.

- (1)M/M/c-Poisson /Exponential multi-server queue with unlimited system capacity (calculates the major measures of effectiveness), (2) M/M/c/K –as M/M/c but limit with system capacity of K and calculates the major measures of performance, (3) M/G/c/c – Poisson input and arbitrary service times at “c” servers, with no waiting room, (4) Optimal number of servers for an M/G/c/c (determines the number of servers to guarantee that overflow/loss probability is no higher than a specified value),(5) M/M/infinity, (6) M/G/infinity, (7)Markov multi server finite source queue without spares, (8)Markov multi server finite source queue with spares, (9)M/E(k)/c- Poisson input to “c”- server queue with Erlang service times and unlimited system capacity, (10)M/D/c, 11)D/M/c, (12)E(j)/M/c–Erlang inter arrivals to multi server queue with Exponential service times

and unlimited capacity, (13)H/M/c – Hyperexponential interarrival times (two versions as balanced and mixture), (14)G/M/c, (15)Bounds for the General multi server G/G/c- gives upper and lower bounds for the system and queue size, waiting times for general multi server queue, (16)General approximation for the G/G/c – calculates approximates for the major effectiveness measures and (17)Optimal number of server for an G/G/c- determines the number of servers to guarantee that average wait time in the queue is below a specified threshold. For easy reference, I made the Table(3) and Table(4) to represent the inputs the user needs to know for each model and the resulting outputs that the user gets by running these models. The acronyms used in the Table3, 4, 5, 6 for the models are given below.

- A M/M/c
- B M/M/c/K
- C M/G/c/c
- D Optimal number of servers for an M/G/c/c
- E M/M/infinity
- F M/G/infinity
- G Markov multilevel, Finite source queue without spares
- H Markov multiserver, Finite source queue with spares
- I M/E(k)/c
- J M/D/c
- K D/M/c
- L E(j)/M/c
- M H/M/c -Mixed
- N H/M/c -Balanced
- O G/M/c
- P G/G/c
- Q General approximation for the G/G/c
- R Optimal number of servers for an G/G/c

In the G/M/c model, the user needs to input interarrival time(t) and the probability a(t) in addition to the parameters given in the table. Other than that, operating “Multiple-server models” is almost the same as operating “Single-server models”.

Table 3: Input for Multiple Server Models for QTS PLUS

INPUT	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
Arrival rate (λ)	V	V	V	V	V	V	V	V	V	V	V	V
Mean service time ($1/\mu$)	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
Number of servers in the system (c)	V	V	V	V	V	V	V	V	V	V	V	V	V	V	..
Maximum # of customers in the system (K)	..	V
Specific time for delay distribution calculation (t)	..	V	V	V
Absolute maximum number of servers	V
Target overflow probability (p_c)	V
Test time for transient analysis	V
Mean interarrival time for a single customer ($1/\lambda$)	V	V	V	V	V
Maximum # of units in the system (M)	V	V
Number of spares (Y)	V
Erlang shape parameter (k)	V	V
Number of alternate phases (n)	V
Proportion of type 1 customers (q)	V
Number of interarrival probabilities	V
Variance of the interarrival times	V	V	V	..
Absolute maximum number of servers	V
Target line delay (Wq)	V
Plot Parameters:																		
Maximum size for probability chart	V	V	V
Total time horizon for probability plotting	V	V	V	V	V	V
Maximum value of variable --																	
whose probability is to be plotted (K)	V	V	V	V	V

Table 4:Output for Multiple Server Models for QTS PLUS

OUTPUT	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
Mean interarrival time ($1/\lambda$)	V	..	V	V	V	V	V	V	V	V	V	V	V
Mean number actually entering service (λ_{eff})	..	V	V	V
Service rate (μ)	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
Average # arrivals in mean service time (r)	V	V
Server utilization (ρ)	V	V	..	V	V	V	V	V	V
Fraction of time all servers are idle (p_0)	V	V	V	V	..	V
Mean number of customers in the system (L)	V	V	..	V	V	V	V	V	V	V	V	V	V	V	V	..	V	..
Mean number of customers in the queue (Lq)	V	V	V	V	V	V	V	V	V	V	V	..	V	..
Mean wait time (W)	V	V	..	V	V	V	V	V	V	V	V	V	V	V	V	..	V	..
Mean wait time in the queue (Wq)	V	V	V	V	V	V	V	V	V	V	V	..	V	..
Ratio of λ to μ (r)	..	V
Probability of the system is full (p_K)	..	V
Expected number turned away per unit time	..	V
Probability that delay in queue > t ($Wq(T>t)$)	..	V	V	V
Individual server utilization (ρ_{eff})	V	V	V
Fraction of time system is full (p_c)	V
Rate of lost customers	V
Expected system size (L)	V
Optimal number of servers (c)	V
Transient mean size at time t	V
Combined effective overall mean arrival rate (λ_{eff})	V	V
Expected number of busy servers (r)	V	V	V	V	..
Fraction of time the server is busy	V	V	V	V	V
Function root[determines answers] (r)	V	V	V	V	V
Fraction of time arrival finds server idle (q_0)	V	V	V	V	V
Probability arrival finds c customers (q_c)	V	V	V	V	V
Expected system size seen by an arrival ($L^{(A)}$)	V	V	V	V	V
Expected queue size seen by an arrival ($Lq^{(A)}$)	V	V	V	V	V
Upper bound on Wq	V
Lower bound on Wq	V
Upper bound on W	V
Lower bound on W	V
Upper bound on Lq	V
Lower bound on Lq	V
Upper bound on L	V
Lower bound on L	V
Optimal number of servers (c)	V
Estimated Wq with c servers	V
Arrival rate (λ)	V	V	V	V	V	V
Charts																		
system size distribution	V	V	V	..	V	V	V	V	V	V	V	V	V
time distribution	V
plot data	V	V	V	V	V	V	V
probability chart
Wq(t) chart	V	V	V	V

3.4.2.5 “Bulk Models, Priority Models, Network Models and Simulator Models” is a special capability of the QTS Plus package that the other packages discussed in this paper do not have. General Inputs and the Outputs of the Bulk model and Priority model are given in Table 5 and Table 6 for easy reference. All these models are operated as the previously discussed ones. The acronyms used in the Tables for the models are given below. Sample outputs for Bulk Models and Priority Models are given in Appendix **(2)-f** and **(2) -g**.

A - $M^K/M/1$ (Markovian, single server fixed size bulk-input, exponential service).

B - $E(j)^K/M/1$ (Erlang inter-arrival times, fixed batch size and exponential service times)

C - $M^X/M/1$ (Markovian, single server bulk input, user input, discrete batch size probabilities)

D - $M^X/M/c$ (Markovian multi server bulk input, discrete batch size probabilities)

E- $M/M^K/1$, Type 1

F – $M/M^K/1$, Type 2

G – Single server, Markovian queue with 2 Priority classes

H – 2 Priority single server Markov model with 2 service rates

I – Single server Markov model with up to 5 Priority classes

J- Multi server Markov model with up to 5 Priority classes

K – Five priority single server Markov model with up to 5 service rates

L- Multi-class, single server Markov model

M – Multi server Markovian queue with 2 priority classes

N- Single server Markov Preemptive model with 2 classes

Table 5 : Input and Output for Bulk Models for QTS PLUS

Input Parameters:	A	B	C	D	E	F
Arrival rate(λ)	v	v	v	v	v	v
Mean service time ($1/\mu$)	v	v	v	v	v	v
Fixed batch size ($K > 1$)	v	..	v	v
Interarrival Erlang shape parameter (j)	..	v
Input batch size (K)	..	v
Number of service channels (c)	v
Maximum number served at a time (K)	v	..
Number served at a time (K)	v
Plot Parameter:						
Maximum size for probability chart	v	..	v	v	v	v
OUTPUT						
Mean interarrival time ($1/\lambda$)	v	v	v	v	v	v
Service rate (μ)	v	v	v	v	v	v
Server utilization (ρ)	v	..	v	v	v	v
Probability of empty system (p_0)	v	..	v	v	v	v
Mean number of customers in the system (L)	v	..	v	v	v	v
Mean number of customers in the queue (L_q)	v	..	v	v	v	v
Mean wait time (W)	v	..	v	v	v	v
Mean wait time in the queue (W_q)	v	..	v	v	v	v
Fraction of time the server is busy (ρ)	..	v
Fraction of time the server is idle (p_0)	..	v
Function root (determines answer)	..	v	v	v
Mean batch size ($E[X]$)	v	v
Batch variance ($Var[X]$)	v	v
Size distribution chart	v	..	v	v	v	v
Plot data	v	..	v	v	v	v

In the following table (Table 6) “v1” indicates that the input value should be the overall mean service time instead of mean service time. In the “Five Priority single server” model (K) the “\$” indicates that both the arrival rate and service rate for the five classes are needed as input. In the “Multi class single server” both arrival rate $\lambda(k)$ and service rate $1/\mu(k)$ should be input for the number of classes (k). The output for this model is given in a separate sheet, other than the outputs given in the Table 6, $\mu(k)$, $p(k)$, $W(K)$, $W_q(k)$, $L(k)$, $L_q(k)$ are also given for each “k” (number of classes).

Table 6 :Input and Output for Priority Models for QTS PLUS

Input	G	H	I	J	K\$	L*	M	N
Mean arrival rate for class 1 (λ_1)	v	v	v	v	v\$	*	v	v
Mean arrival rate for class 2 (λ_2)	v	v	v	v	v\$	*	v	v
Mean arrival rate for class 3 (λ_3)	v	v	v\$	*
Mean arrival rate for class 4 (λ_4)	v	v	v\$	*
Mean arrival rate for class 5 (λ_5)	v	v	v\$	*
Mean service time ($1/\mu$)	v	..	v1	v1	..	*	v	v1
Mean service time for class 1 ($1/\mu_1$)	..	v	*
Mean service time for class 2 ($1/\mu_2$)	..	v	*
Number of servers (nc)	v	v	..
Number of classes(k)	v
Plot Parameter:								
Maximum size for probability chart	v	..	v	v	..	.	v	..
Output								
Overall arrival rate (λ)	v	v	v	v	v	v	v	v
Mean interarrival time ($1/\lambda$)	v	v	v	v	v	v	v	v
Service rate (μ)	v	v	..
Overall service rate (μ)	..	v	v	v	v	v
Overall mean service time ($1/\mu$)	..	v
Server utilization (ρ)	v	v	v	v	v	..	v	v
Probability of an empty system (p_0)	v	v	v	v	v	v	..	v
Mean number of customers in the system (L)	v	v	v	v	v	v	v	v
Mean number of customers in the queue (Lq)	v	v	v	v	v	v	v	v
Mean wait time (W)	v	v	v	v	v	v	v	v
Mean wait time in the queue (Wq)	v	v	v	v	v	v	v	v
Average # arrivals during mean service (r)	v	v	..
Expected number being served at any time (ρ)	v
Fraction of time system is idle (p_0)	v	..
Priority Class #1:								
Expected time in the system (W1)	v	v	v	v	v	*	v	v
Expected waiting time in the queue (Wq1)	v	v	v	v	v	*	v	v
Expected number in the system (L1)	v	v	v	v	v	*	v	v
Expected number in the queue (Lq1)	v	v	v	v	v	*	v	v
Priority Class #2:								
Expected time in the system (W2)	v	v	v	v	v	*	v	v
Expected waiting time in the queue (Wq2)	v	v	v	v	v	*	v	v
Expected number in the system (L2)	v	v	v	v	v	*	v	v
Expected number in the queue (Lq2)	v	v	v	v	v	*	v	v
Priority Class #3:								
Expected time in the system (W3)	v	v	v	*
Expected waiting time in the queue (Wq3)	v	v	v	*
Expected number in the system (L3)	v	v	v	*
Expected number in the queue (Lq3)	v	v	v	*

Table 6. contd

Output	G	H	I	J	K\$	L*	M	N	O
Priority Class #4:									
Expected time in the system (W4)	v	v	v	*	v
Expected waiting time in the queue (Wq4)	v	v	v	*	v
Expected number in the system (L4)	v	v	v	*	v
Expected number in the queue (Lq4)	v	v	v	*	v
Priority Class #5:									
Expected time in the system (W5)	v	v	v	*	v
Expected waiting time in the queue (Wq5)	v	v	v	*	v
Expected number in the system (L5)	v	v	v	*	v
Expected number in the queue (Lq5)	v	v	v	*	v
Charts									
System size chart	v	..	v	v

“M/M/c queues in series”, “Open Jackson network model”, “Two class open Jackson network model”, “Closed Jackson networks :Buzen Algorithm” and “Closed Jackson network:MVA” comes under “Network models” . In all of these models user needs to input number of nodes, μ or $1/\mu$ with the other inputs. Output appear in a separate worksheet where the user can toggle between these input and output pages. Some of the sample outputs given in the package is given as an example in Appendix (2) -h.

3.4.3 MC Queue

To use this package the user needs to select “OR STAT” from the extracted files. Then from the front page user needs to select “Modules”. “Modules” contain two parts named, “Queue” and “Markov chains”.

3.4.3.1 Under the module “Queue” this package is capable of handling M/G/1 , D/G/1 , M/M/c , M/D/c , G/M/c , Finite capacity queues , Finite source queues and Transient M/M/1 queues. To use this module the user needs to select the required model. After selecting the model, the user can input the required input values by typing it in the relevant boxes and the out put will appear in a separate window by the side of the input values. In this package for every model the performance measures are calculated under the assumption that the service time is normalized. ($E(S) = 1$). Therefore when we compare the values with the values of the other packages , we need to convert the output of this package accordingly.

3.4.3.1.1 For the M/G/1 Model there are three choices (Erlangian , Coxian -2[gamma normalization], coxian -2 [balanced means]) for the service time where the user can select one according to the requirement. The input parameters are CS2 (squared coefficient of variation – the ratio of the variance and the squared mean) and Rho ($\lambda * E(s)$) . Before running the program, the user can decide which output is required and can choose it from the output selection part. The user has five choices namely, “standard statistics” , “waiting time probabilities” , “waiting time percentiles” , “state probabilities” and “graphs”. Standard statistics gives the steady state P_{wait} , P_{busy} , $E(W_q)$, $E(L_q)$, $Std(W_q)$, and $Std(L_q)$. The option “Waiting time probabilities” gives the “Standard statistics” and the $Pr(W_q > t)$ for specified values of “t” (user needs to input the values of “t” if this option selected) . For the option “Waiting time percentiles”, the user will get the “Standard statistics” and calculate “x” such that $P\{W_q \leq x\} = p$ for a given values of “p”. The option “State probabilities” gives “Standard statistics” plus the probabilities $P\{L = n\}$ and $P\{L > n\}$ for specified values of “n” (for this option, the user needs to input values of “n”). The option “Graph” gives the logarithmic plot of the conditional probabilities $P\{W_q > t | t > 0\}$ against “t” and $P\{L > n | n > 0\}$ against “n”.

3.4.3.1.2 The D/G/1 model has the same input as the above M/G/1 model except the D/G/1 model has another option for the service time, “Pure Erlangian”. This option requires input $1/CS2$ instead of CS2 and $1/CS2$ should be an integer. The output is as same as the M/G/1 and gives “Time average state probabilities” and “Customer average state probabilities” in addition to the M/G/1 output. The “Graph” option gives the logarithmic graph of the conditional probability $P\{W_q > t | t > 0\}$ against “t”.

3.4.3.1.3 The models M/M/c , M/D/c , D/M/c and G/M/c give the same output as the M/G/1 model and for these models, the user needs to input values for the “number of servers” and Rho. In the D/M/c model, “Time average state probabilities” and “Customer average state probabilities” are provided only for “Pure Erlangian service time”. For this model, 3 graphs are available : probabilities $P\{W_q > t | t > 0\}$ Vs “t” , $P\{L > n | n > 0\}$ Vs “n” and probability $P\{W_q > t | t > 0\}$ Vs “t” }.

3.4.3.1.4 “Finite capacity queues” are capable of handling the modules M/M/c/c+N , M/D/c/c+N , D/M/c/c+N. The user can select one of these models and input the required values. For all these models, the input values are the same: c- the number of servers ,

ρ , N – maximum number of customers allowed to wait in queue. For the models $M/M/c/c+N$ and $M/D/c/c+N$, the user has four options for the output, namely “Standard statistics”, “Waiting time probabilities”, “State probabilities” and “Graphs”. “Standard statistics” gives the steady state quantities of P_{rej} (this is describe in the package under “Help” the probability that a customer finds upon arrival all of the waiting places occupied and thus is rejected -loss probability), P_{wait} –delay probability, P_{busy} , $E(W_q)$, $Std(W_q)$, $E(L_q)$, $Std(L_q)$ and Utilization (this is define in “Help” as the long run fraction of time a given server is busy). For $D/M/c/c+N$ the user can get “Time average state probabilities” and “Customer average state probabilities” in addition to the above four options.

3.4.3.1.5 “Finite source queues” are also called the Machine repair model. The user has two options for the service time:exponential and deterministic. Input parameters for this model are defined in the “Help” menu as follows: c - number of servers, M - number of sources, ν - the rate at which the source generates service requests whenever the source is idle (failure rate of an operating machine). Output for this model is the same as $M/G/1$ except for the “Graph” option this model gives three graphs, as in the $D/M/c$ model.

3.4.3.1.6 Under “Transient $M/M/1$ queue” the package calculates transient performance measures. Generally, the input parameters are ρ , Initial state (i), End state (j) and “ t ”(time period). The user has five options for the output:“Table of moments of queue size”(which gives the quantities $p(ijt)=P\{L(i,t)= j\}$ where $P\{L(i,t)= j\}$ means the probability that “ j ” customers in the system at time “ t ” given that “ i ” customers are present at time “0”(initial state) , $p(i,j,t) = P\{L(i,t)\leq j\}$, $EL(i,t) =$ expected value of $L_q(i,t)$ where $L_q(i,t)$ means the number of customers in the queue at time “ t ” given that “ i ” customers are present at time “ t ”, $Std L_q(i,t) =$ standard deviation of $L_q(i,t)$), “Graph state probabilities” ($p(i,j,t) = P\{L(i,t) = j\}$ Vs “ t ”), “Graph of cumulative state probabilities” ($P(i,j,t) = P\{L(i,t) \leq j\}$ Vs “ t ”), “Graph of mean queue size” ($EL_q(i,t)$ Vs “ t ”), And “Graph of standard deviation of queue size” ($Std L_q(i,t)$ Vs “ t ”).

3.4.3.2 The “Markov chains” module is capable of handling both discrete-time Markov chains and continuous-time Markov processes having a finite discrete , the state space. For both cases user needs to input the transition matrix and there is an option of imposing a cost structure on the Markov chain. For this module, the user has four options for the

output: “Chain structure”(determines for each state whether the state is recurrent or transient), “Steady state behavior”(if the Markov chain has a single recurrent class, the steady state probabilities are given), “Transient behavior”(calculates the transient probabilities) and “First – passage time probability”. There is special requirement for “Steady state behavior” in that this module requires that the Markov chain have single recurrent class. This is a special facility of “MC Queue”. Appendix (3) gives a general output.

3.4.4 Quick Q

To use this package, the user needs to unzip the zipped file, extract the files in it, select “Q” and press “use Quick Q”. By doing that user can go to the window where all the calculations are being done. This version of “Quick Q” is currently capable of handling the models M/M/1 , M/M/s , M/D/1 , D/D/1 , D/D/s , M/G/1. But in this package, the user cannot identify these models easily as in previously described packages. Before entering the parameters, the user needs to decide which model is needed and then input the values in the general input boxes accordingly. By selecting the “Entrance distribution” (two options are Exponential and Constant), “Server distribution”(three options Exponential, Constant and General) and by changing the number of servers, the user can have different models as stated above. To get the average queue size, average system size and the server utilization, the user needs to input the average value for the entrance distribution and the server distribution. Then the user needs to choose “Calculate Now” from the “option” menu. If the current queue type described by the input values is solvable, Quick Q will indicate it on top of one of the boxes by displaying “Solvable” in green letters. But if it indicate “Not Solvable” that does not mean Quick Q will not do any calculations. It will calculate as many parameters as it can. The calculated fields will be grayed and the user can identify them easily. For this Quick Q package the fields such as entrance distribution, server distribution and the number of servers must always be filled up.

3.4.5 QTP

This is really an Excel spread sheet where the user can make his/her own worksheet using the pre-defined functions of QTP. There are two types of pre-defined functions in QTP Tool Pack :functions whose names begin with “QTPMMS_” and functions whose names begin with “QTPGGS_”.

These two function types differ, based on their assumptions. For “QTPMMS” it is assumed that arrivals are from a stationary Poisson distribution, customers wait in a single queue, service time is exponentially distributed and customers are served in a first come first serve basis. For “QTPGGS” it is assumed that interarrival times are independent, but can follow any distribution, the queue is assumed to have unlimited capacity, the service time can follow any distribution and customers are served on a first come first serve basis. For a first time user, it is advisable to do the two tutorials first to get a knowledge about how to use this package. The tutorials give a very good description to the user on how to make a worksheet to do the required calculations. The list of functions given in the package is in Appendix (4)-a. This package is capable of giving error messages whenever an error occurs (some time error messages generated by Excel can appear). These messages appear when the pre-defined function is entered with inappropriate arguments. A list of error messages is given in the “Help” menu. But it states that the absence of an error message should not be taken as a guarantee that the values are accurate. A general worksheet for QTP is given in Appendix (4)- (b).

The capabilities of the five packages, JPQ, QTS Plus , Quick Q , MC Queue , QTP are summarized in the Table(7) below.

Table 7: Summary of Statistical capabilities

Capabilities	Queueing softwares				
	JPQ	QTS Plus	Quick Q	MC Queue	QTP
M/M/1	X	X	X	X	X
M/M/1/K	X	X	~	X	X
M/M/C	X	X	X	X	X
M/M/inf	X	X	X	~	X
M/M/c/K	X	X	~	X	X
M/M/c/N	X	X	~	X	~
M/M/c/N/Y	X	X	~	X	~
Transient M/M/1	~	X	~	X	~
Queue with state dependent service	~	X	~	~	~
M/G/1	~	X	X	X	X
M/D/1	~	X	X	X	X
M/E(k)/1-Erlang service time	~	X	~	X	~
D/M/1	~	X	~	X	X
D/D/1	~	~	X	~	X
D/D/c	~	~	X	~	X
E(j)/M/1- Erlang interarrival time	~	X	~	~	~
G/M/1	~	X	~	X	X
G/G/1	~	X	~	~	X
E(j)/E(k)/1	~	X	~	~	~
M/D/c	~	X	~	X	X
D/M/c	~	X	~	X	X
M/G/c/c	~	X	~	~	X
E(j)/M/c	~	X	~	~	~
H/M/c	~	X	~	~	X
G/M/c	~	X	~	X	X
Bulk models					
M ^[K] /M/1	~	X	~	~	~
E(j) ^[K] /M/1	~	X	~	~	~
M ^[X] /M/1	~	X	~	~	~
M ^[X] /M/c	~	X	~	~	~
M/M ^[K] /1:Type 1	~	X	~	~	~
Probability Calculators					
Poisson Calculator	~	X	~	~	~
Erlang Calculator	~	X	~	~	~
Mixed Exponential Calculator	~	X	~	~	~
Markov Chain Models					
Discrete-Time	~	X	~	X	~
Discrete-Time(Transient analysis)	~	X	~	X	~
Continuous Time	~	X	~	X	~
Priority models	~	X	~	~	~
Network models	~	X	~	~	~

Chapter 4

RESULT TESTING AND FURTHER FINDINGS

4.1 RESULT TESTING

WeI tried to find some benchmarks to assess the quality of the algorithms implemented in each package, but we couldn't find any for Queueing models. Instead we tested the behavior of each package at some extremes (ρ close to 1). We compared the output results with those we calculated using MAPLE (Table 9).

Some of the outputs are tabulated below in Table (8), for M/M/1 model.

Table 8:Output for model M/M/1, from the five packages

λ^*	8.9	8.99	8.9955	9	8.999999
μ^*	9	9	9	9	9
JPQ - ρ	0.989	0.999	1	-	1
L	89.0	899.0	1999.0	-	8999999.011
L_q	88.011	898.001	1998	-	8999998.011
W	10.0	100.0	222.222	-	1000000.001
W_q	9.889	99.889	222.111	-	999999.89
QTS ρ	98.89%	99.89%	99.95%	-	100%
L	89	899	1999	-	8999999.011
L_q	88.01111111	898.0011111	1998.0005	-	8999998.011
W	10	100	222.222222	-	1000000.001
W_q	9.888888889	99.88888889	222.1111111		999999.8901
MCQueue ρ	0.988888*	0.998888*	0.99950008*	1*	0.99999988*
L	-	-	-	-	-
L_q	88.004	897.28	-	-	-
W	-	-	-	-	-
W_q	9.88811111	99.80888889	-	-	-

Table 8:contd.

QuickQ	98.89%	99.89%	99.95%	-	100%
L	89.0000	899.000	1999	-	899999
L _q	88.0111	898.001	1998.00	-	899999
W	10.0000	100.000	222.222	-	999999.
W _q	9.8888	99.8888	222.111	-	999999.
QTP	.988888889	.998888889	.999500	1	.99999989
L	89.00000	899.000000	1999.0000	-	-
L _q	88.01111111	898.00111111	1998.000500	-	-
W	10.000000	100.00000	222.2222222	-	-
W _q	9.88888889	99.88888889	222.1111111	-	-

In Table 8 , “*” indicates input values that are not calculated by the relevant package. Calculated values for ρ , L , L_q ,W , W_q using Maple are tabulated below in Table (9). For the calculations I used the equations given in the section (2) for the M/M/1 model.

Table9:Calculated values for model M/M/1 using MAPLE

λ	8.9	8.99	8.9955	9	8.999999
μ	9	9	9	9	9
ρ	.9888888889	.9988888889	.9995000000	1	.999999888
L	89.00000000	899.0000000	1999.000000	-	8999999.000
L _q	88.0111112	898.0011201	1998.000500	-	9.0008980x10
W	10.00000001	100.0000010	222.2222222	-	1000100.010
W _q	9.888888889	99.88888889	222.1111111	-	999999.8889

The values that of ρ , L, L_q,W, W_q are almost the same for all packages except for some rounding errors when we use the packages JPQ, QTS, Quick Q, QTP and Maple. I

assume values for L_q , and W_q we get using MC Queue are slightly different due to the assumption that $E(S) = 1$. This means the input for ρ slightly inaccurate.

When $\lambda = 8.9955$ and $\mu = 9$, for M/M/1 model $\rho = 0.9995$. But the package JPQ it writes this value as one(1) but internally keeps the correct value and gives correct values of L , L_q , W , W_q . When $\rho = 1$ we cannot get values of L , L_q since $1 - \rho = 0$. QTS Plus, Quick Q and QTP print the exact value of 0.9995. When $\lambda = 8.999999$ and $\mu = 9$, we get the value of ρ as .999999888 and although only MC Queue and QTP print the correct value of ρ , these two packages internally round it off to be equal to “1” and do not give answers for L , L_q , W , W_q . On the other hand JPQ, QTS Plus, Quick Q print $\rho = 1$ but these packages calculate values of L , L_q , W , W_q using $\rho = .999999888$. In MC Queue the case is different because in this package, the user must input the value of “ ρ ” and in that package there is a condition that ρ should be less than 0.999 in order to calculate the required values. Therefore it is impossible to check the outcome when $\rho > 0.999$.

4.2 FURTHER FINDINGS

If any of the users, who use these packages are interested to improve their existing systems, can make use of these packages to do that. Different system measurements were studied using QTS PLUS for fixed service rate with varying entrance rate and varying service rate with varying entrance rate for different single server models.

Expected number of customers in the system and the mean waiting time in the queue was obtained for fixed ρ (by varying, service rate and arrival rate). The output was used to plot the following two graphs.

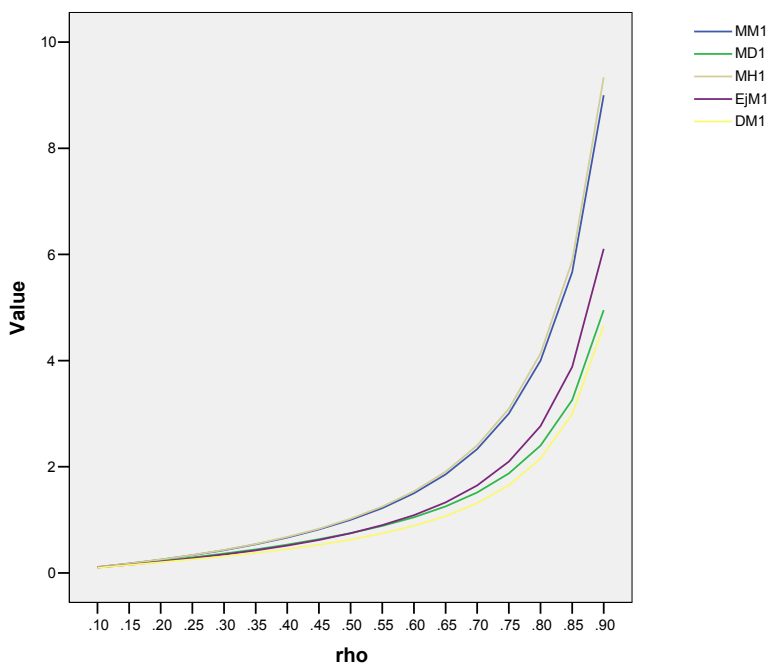


Figure 1:Expected number of customers in the system VS rho

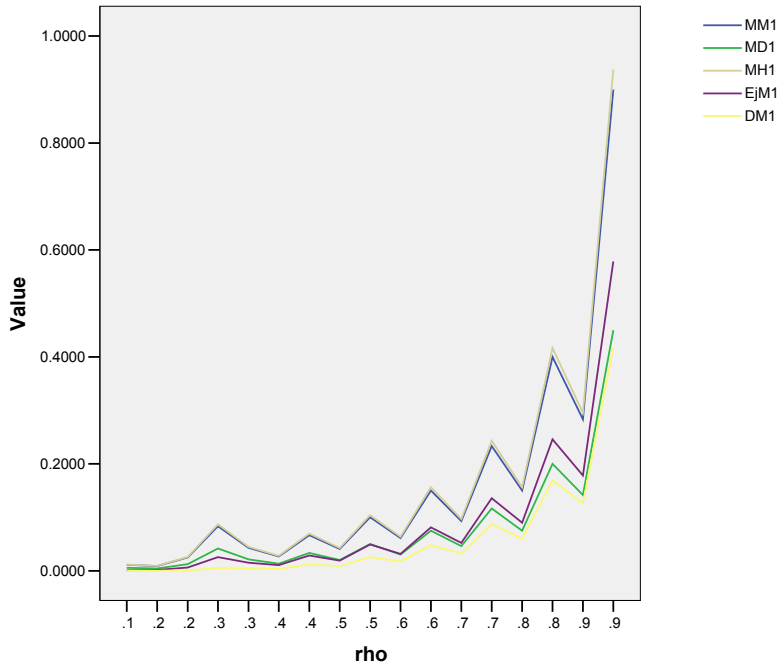


Figure 2: Mean waiting time in the queue Vs rho

From the graph of “Expected number of customers in the system Vs rho” ,Figure 1, we can see the model D/M/1 has the minimum number of customers in the system than all the other models. That is when the arrival time is discrete (scheduled arrival times) the system has the minimum number of customers at a given time. So, the user can use this information to improve the existing system.

From the graph of “Mean waiting time Vs rho”, Figure 2, I understand that the waiting time is not a function of “rho”. Therefore the graph , does not clearly indicate what kind of a changes to be done to the existing system to improve the performances.

Instead of varying service rate and varying arrival rate I used varying arrival rate with fixed service rate to get the expected number of customers and the mean waiting time in the queue. The output was used to plot the following two graphs.

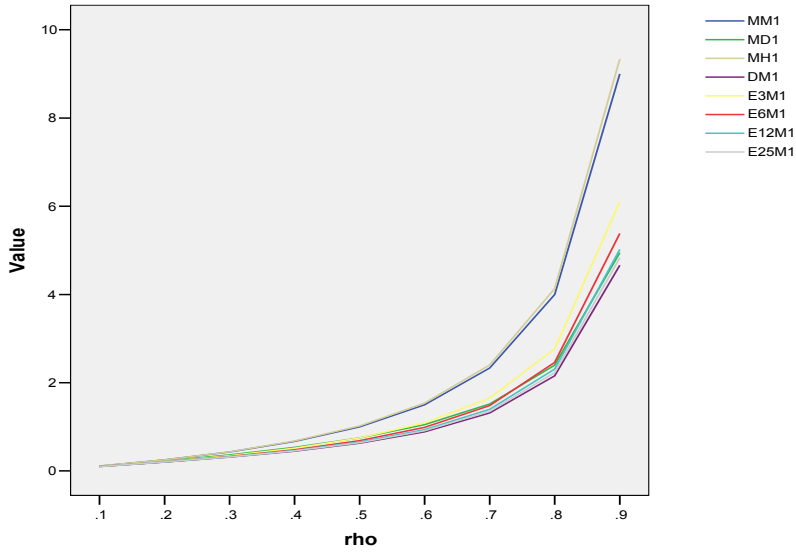


Figure 3: Mean number of customers in the system Vs rho for fixed mu (.1).

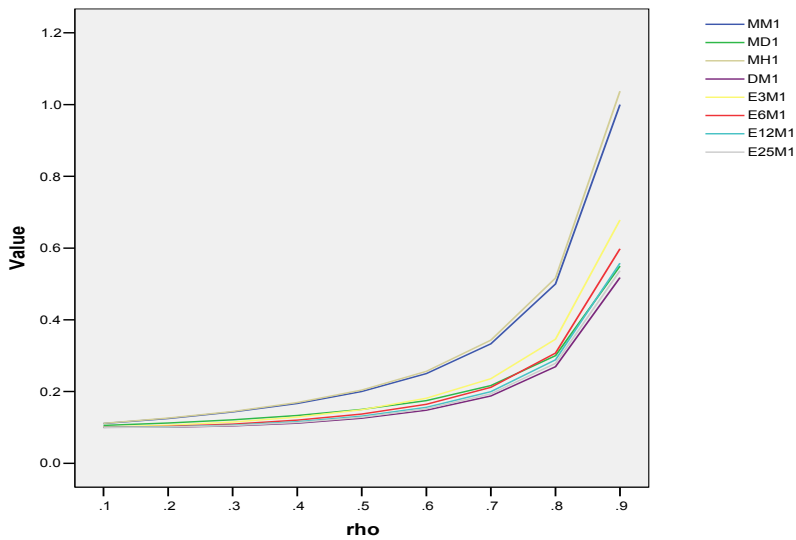


Figure 4: Mean waiting time in the system for fixed mu(.1).

Both of these graphs, Figure 3 and Figure 4, clearly indicate that the model D/M/1 has the least values for the mean number of customers in the system and the mean waiting time in the queue.

Chapter 5

Discussion and Conclusion

5.1 DISCUSSION

JPQ

This package is very easy to use since it has straight-forward menus and fewer options. But this package has fewer capabilities than the other packages. The output is easily viewed on the screen but since the package does not allow Windows cut and paste, it is impossible to get a hardcopy of the output by any method. The Java applet does not allow printing and copying and even the “shift/print screen” is not working for this package. This is the biggest drawback of this package. The other drawback is that the user cannot go back to the earlier models. It is hard to compare the output values of this package with output of other packages since one needs to write down the output of this package. The user can clearly identify input and output values. When the first Java applet page appear it advices to down load Java-Plug-in to operate this software. But in my case, although I downloaded the Java- Pulg-in, I did not need this one, to operate this package. The other problem I encountered was, I can not operate this JPQ in some other computers.

QTS Plus

All the models in QTS Plus are in Excel work sheets and the user has the privilege of printing, editing, saving and other facilities available for Excel. Since this package is in an Excel environment, the user can easily go back to earlier models. Input and output values are clearly defined and the package is easy to use. This package has more capabilities than some other packages and it has special capabilities that are unique to this package only.

MC Queue

In this package, the user has to input the value of $\rho = (\lambda/\mu)$ instead of usual λ and μ . This is a bit inconvenient since the user has to do the calculation first and feed it to the computer. But in other way this package is good since within the package (and in almost every other one discussed in this paper) it will do the rounding off and the user does not need to round off. Using this package, the user will not get values of “L” and “W”. Since this package has an assumption $E(S) = 1$, to get the value of W_q , the user must do another calculation $E(W_q)/\mu$. This is also cumbersome work. Another drawback is that this package does not accept values greater than 0.999 for ρ , so it is impossible to check behavior in extreme cases like the other packages. But MC Queue facilitates the printing of outputs, except the graphs. This package is easy to use and has special capabilities unique to it, like giving the state probabilities when user gives the state.

Quick Q

The best part of this Quick Q is that it will calculate all of the values the package can, based on the parameters that have been entered so far (unlike JPQ). Another good thing is that, at any time the user can change or delete any of the values entered and if so, this package will then recalculate all other values accordingly. But, the user can not toggle between different work pages like in QTS. The user needs to close the existing model to get a new one, and hence the user can not go back to the earlier pages easily. The current version of Quick Q does not allow printing, copying or editing. Therefore it was hard to include an example in the appendix. Quick Q did not clearly specify which parameters the user needs to input like the other packages. The way of giving the output is not as straight forward as other packages. This package does not have as many capabilities as QTS or MC Queue.

QTP

Since QTP is in an Excel environment and the user makes his/her own worksheet, it is very easy to print the output and graphs. In this package, the user needs to remember the pre-defined functions. Therefore in the beginning, the user faces some difficulties but once the user makes his/her own worksheet according to the user's requirements, the user can overcome those difficulties. At a glance, this package seems to be not user friendly, but once the user starts working in it, this package is as good as the other packages QTS, MC Queue. Using the pre-defined functions user can make a work sheet which is capable of doing as many calculations as QTS and MC Queue.

5.2 CONCLUSIONS

The five packages, JPQ, QTS PLUS, QTP, QUICK Q, and MC QUEUE have different capabilities. QTS PLUS is the most featured package and the other packages also intended to provide some help to the user. When we compare the features such as installation, hardware requirements, documentation, available procedures, accuracy, user friendliness, reliability, technical support and capabilities, of these five packages, we can recommend QTS PLUS is the best among those five. But anyhow the founders of these five packages, JPQ, QTS PLUS, MC QUEUE, QTP and QUICK Q, spent their precious time to make these packages and all these packages can be download without any cost. There are so many queueing software packages available now. I randomly picked these five packages and I enjoyed working with each one of them.

REFERENCES

Donald Gross and Carl M. Harris, “Fundamentals of Queueing Theory”, third edition, 1998 .

Sheldon M. Ross, “Introduction to Probability Models”, fourth edition, 1989.

Robert A. Oster (2003), “An Examination of Statistical Software Packages for Categorical Data Analysis Using Exact Methods – Part-II,” *The American Statistician* , 57, 201-213.

Herman Callaert (2003) , “Comparing Statistical Software Packages: The Case of The Logrank Test in StatXact” *The American Statistician*, 57, 214-217.

Robert A. Oster (1998) , “An Examination of Five Statistical Software Packages for Epidemiology, ” *The American Statistician*, 52, 267-280

B. D. McCullough (1998) , “Assessing the Reliability of Statistical Software : Part I,” *The American Statistician* , 52, 356-366

Robert M. Hamer (1981) , “Papers That Evaluate Computer Programs,”*The American Statistician* , 35, 264

Models Overview in the web site <http://www.geocities.com/qtsplus>

Appendix (1) – a

Notations used for JPO package

- λ - Mean arrival rate
- μ - Mean service rate
- c - Number of servers
- k - Buffer size
- N - Number of machines in the system
- Y - Number of spare machines (stand by machines)
- ρ - Traffic intensity
- P_n - probability of having “n” customers in the system
- λ_e - Effective arrival rate
- X - Number of customers in the system at a random time
- L - Expected system size ($L = E(x)$)
- σ_X - Standard deviation of X
- X_q - Queue size at random time
- L_q - Expected system size ($L = E(X_q)$)
- σ_{X_q} - Standard deviation of X_q
- T - Waiting time in the system of a random customer
- W - Expected waiting time in the system
- σ_T - Standard deviation of T
- T_q - Waiting time in queue of a random customer
- W_q - Expected waiting time in queue
- σ_{T_q} - Standard deviation of T_q

Appendix (1) – b

General output for model M/M/1 for JPO

$n=2$, $t=2$, $t_q=1.5$

λ	8.5	5	9
μ	9	8	8
ρ	0.944	0.625	--
L_q	16.056	1.042	--
σ_{X_q}	17.44	1.852	--
W_q	1.889	0.208	--
σ_{T_q}	1.997	0.309	--
L	17.0	1.667	--
σ_X	17.493	2.108	--
W	2.0	0.333	--
σ_T	2.0	0.333	--
P(X>n)	0.892	0.3906	--
P(T>t)	0.3679	0.0025	--
P($T_q > t_q$)	0.4461	0.0069	--
Error msg	no	no	Yes**

** error message - Rho >1, can not converge to a solution

Appendix (1)- c

General output for model M/M/1/K for JPQ

K	4	10
λ	8.9	3
μ	9	5
ρ	0.989	0.6
P_K	0.196	0.0023
λ_e	7.16	2.993
L_q	1.182	0.861
W_q	0.165	0.288
L	1.978	1.46
W	0.276	0.488

Appendix (1)- d

General output for model M/M/C for JPQ

(For C = 3 , n = 4 , t = 2 , t_q = 1.5)

λ	8.9	12	9
μ	9	4.0001	3
ρ	0.33	1	--
L_q	0.043	39998.11	--
σ_{X_q}	0.29	40000.5	--
W_q	0.0050	3333.176	--
σ_{T_q}	0.023	3333.333	--
L	1.032	40001.111	--
σ_X	1.077	40000.5	--
W	0.116	3333.426	--
σ_T	0.113	3333.333	--
$P(X \geq n)$	0.0292	0.9999	--
$P(T > t)$	0.0	0.9994	--
$P(T_q > t_q)$	0.0	0.9995	--
Error message	- no -	- no -	- yes** -

** error message - Rho > 1, can not converge to a solution.

Appendix (1)-e

General output for model M/M/inf for JPQ

λ	4	5	4	3.99	3.999	4.00001
μ	5	4	4	5	5	5
L_q	0	0	0	0	0	0
W_q	0	0	0	0	0	0
L	0.8	1.25	1.0	0.798	0.8	0.8
W	0.2	0.25	0.25	0.2	0.2	0.2

Appendix (1)-f

General output for model M/M/C/K for JPQ

C	5	5	5	5	5
K	20	4	4	5	5
λ	20	20	20	20	20
μ	5	5	4	4	5
ρ	0.8	--	1	1.0	0.8
P_K	0.0040	--	0.054	0.222	0.172
λ_e	19.921	--	18.92	15.566	16.566
L_q	1.935	--	6.483	0.222	0.275
W_q	0.097	--	0.343	0.014	0.017
L	5.919	--	11.213	4.113	3.588
W	0.297	--	0.593	0.264	0.217
Error msg	no	Yes*	no	no	no

*error msg – invalid format or number, Please reenter, C>K not feasible.

Appendix (1)-g

General output for model M/M/C/C for JPQ

C	2	2	5	5
λ	4	5	20	20
μ	5	4	4	5
ρ	0.4	0.625	0.25	0.8
P_C	0.151	0.258	0	0.199
λ_e	3.396	3.711	20.0	16.019
L_q	0.0	0.0	0.0	0.0
W_q	0.0	0.0	0.0	0.0
L	0.679	0.928	5.0	3.204
W	0.2	0.25	0.25	0.2

Appendix (1)-h

General output for model M/M/C/N for JPQ

C	5	5	5	2	2
N	4	5	5	2	2
λ	20	20	20	4	5
μ	5	4	5	5	4
λ_e	--	16.667	20.0	4.444	4.444
L_q	--	0.0020	0.0	0.0	0.0
W_q	--	0.0	0.0	0.0	0.0
L	--	4.167	4.0	0.889	1.111
W	--	0.25	0.2	0.2	0.25
ERROR MSG	YES**	no	no	no	no

** -Error message – Invalid format or number, Please reenter, $C > N$, not feasible.

Appendix (1)-i

General output for model M/M/C /N/Y for JPQ

N	4	4	5	4	5
C	5	5	5	5	5
Y	3	7	0	0	0
λ	20	20	20	20	20
μ	5	5	5	5	4
λ_e	24.215	24.993	20.0	16.0	16.667
L_q	0.941	4.733	0.0	0.0	0.0
W_q	0.039	0.189	0.0	0.0	0.0
L	5.784	9.732	4.0	3.2	4.167
W	0.239	0.389	0.2	0.2	0.25

Appendix (2) – a

Probability calculator in QTS Plus

MIXED EXPONENTIAL PROBABILITY COMPUTATIONS

Note that we allow both regular exponential mixtures, which are convex combinations of negative exponential functions, as well as generalized mixtures where the proportionality constants may be negative although still summing to 1.

Clear Old Data

Input Parameters:

p_i	θ_i
1.	0.25
-1.	0.333333
1.	0.5

Plot Parameters:

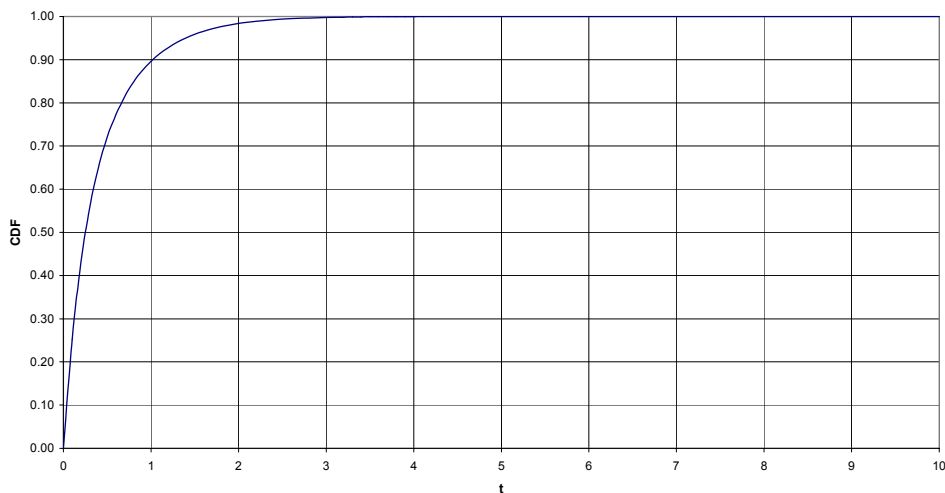
Maximum print/plot value 10

Solve

Results:

Mean value 0.416667
Second moment about zero 0.402778
Variance 0.229167
Coefficient of variation 1.148913

Mixed Exponential Distribution Function



Appendix (2) – b

Exponential and Erlang distribution estimation in QTS PLUS

**EXPONENTIAL AND
ERLANG DISTRIBUTION
ESTIMATION**

This routine takes a random sample of input data and statistically tests whether the values could be considered either as Erlang or exponentially distributed. To start a new problem, clear out any old data by pressing the button "Clear Old Data".

Clear Old Data

Put data below into two columns of approximately equal length
Enter data and then press "Solve" button.

Output		
Observations in 1st column:	13	
Observations in 2nd column:	12	
<i>Test for exponential</i>		
Test Statistic	1.607581	
Estimated Mean	13.62	
For Test Level = 10%,	YES	
For Test Level = 5%,	YES	
For Test Level = 1%,	YES	
<i>Test for Erlang</i>		
Test Statistic	3.039689	
Estimated Mean	13.62	
Estimated Shape Parameter	2	
For Test Level = 10%,	NO	
For Test Level = 5%,	NO	
For Test Level = 1%,	NO	

			Column1	Column2	
			27.6	9.6	
			28.9	8.6	
			8.8	15.7	
			16.6	14.9	
			13.3	11.3	
			8.3	21.3	
			12.8	6.1	
			45.3	6.3	
			12.6	5.6	
			6.8	7.7	
			12.9	8.3	
			9.8	8.8	
			12.6		

Appendix (2) – c

OUTPUT FOR Markov chain solver in QTS PLUS

1) Discrete time Markov chain

MARKOV CHAIN

To start a new problem, enter number of states and press button.

Number of States:

Enter transition probabilities, then press "Solve" button.

Solution

p0	0.230769
p1	0.192308
p2	0.576923

Enter P Matrix below

0.333333	0.333333	0.333333
0.5	0.	0.5
0.1	0.2	0.7

2) Discrete time Markov chain transient analysis

TRANSIENT RESULTS FOR FINITE MARKOV CHAINS

To start a new problem, enter number of states.

Number of States:

Enter number of transitions required.

Number of Transitions:

Enter transition probabilities, then press "Solve" button.

Enter P Matrix below

0.333333	0.333333	0.333333
0.5	0.	0.5
0.1	0.2	0.7

Answer Matrix

(below is the matrix of transition probabilities for 5 steps of the chain)

0.232345	0.193636	0.574019
0.233397	0.190862	0.575741
0.229263	0.192258	0.578479

Appendix (2) – c-contd.

3) Continuous time Markov chains

FINITE CONTINUOUS-TIME MARKOV CHAIN

To start a new problem, enter number of states and press button.

Number of States:

Press "Solve" button to obtain the steady-state probabilities.

Solution

p0	0.384615
p1	0.230769
p2	0.384615

Enter Q Matrix below

-1.0	0.6	0.4
1.0	-2.0	1.0
0.4	0.6	-1.0

4) Finite birth and death model

FINITE BIRTH-DEATH PROCESS

To start a new problem, enter number of states and press button to setup new problem.

Number of Rates:

Enter birth and death rates below, then press "Solve" button.

Solution

Mean	0.428571
lam_eff	1.285714

State	p _n
0	0.714286
1	0.178571
2	0.071429
3	0.035714

Birth Rates Death Rates

1.	***N/A***
2.	4.
3.	5.
N/A	6.

Appendix (2) – c-contd.

5) Finite birth –death model

To start a new problem, enter the number of states needed.

Number of Birth/Death Rates:

Highest State Probability Wanted:

Enter the birth and death rates below, and then press the "Solve" button.

	Solution
Mean	3.77794
lam_eff	1.78060

State	p_n
0	0.072792
1	0.072792
2	0.109188
3	0.218375
4	0.181980
5	0.164225
6	0.086023
7	0.045060
8	0.023603
9	0.012363
10	0.006476

Birth Rates	Death Rates
1.	***N/A***
3.	1.
2.	2.
1.	1.
3.7	1.2
1.1	4.1
N/A	2.1

Appendix (2) – d

Simple Markov queue with state dependent service in QTS PLUS

System with single server having one switch point, FIFO, unlimited capacity.

Input Parameters:

Mean interarrival time for a single customer ($1/\lambda$)	30.
First mean time to complete service ($1/\mu_1$)	40.
Second mean time to complete service ($1/\mu_2$)	20.
Switch number of customers	3

Plot Parameter:

Maximum size for probability chart	20
------------------------------------	----

Results:

Mean arrival rate (λ)	0.033333
First mean service rate (μ_1)	0.025
Second mean service rate (μ_2)	0.05
Overall mean service rate (μ_{eff})	0.038333
First traffic intensity (ρ_1)	1.333333
Second traffic intensity ($\rho_2 < 1$)	0.666667
System utilization ($\rho_{eff} < 1$)	86.96%
Fraction of time server is idle (p_0)	0.130435
Expected queue size (L_q)	2.086957
Expected system size (L)	2.956522
Expected waiting time in the queue (W_q)	62.608696
Expected waiting time in the system (W)	88.695652

Probability Table:

Size	prob(n)	Cumulative Prob(n)
0	0.130435	0.130435
1	0.173913	0.304348

Appendix (2) – e

Erlang input to single Erlang server

After entering the input parameters, press the "Solve" button.

Please note that time to solve is dependent on k, larger values require longer solution times.

Input Parameters:

Interarrival Erlang shape parameter (j)	1
Service Erlang shape parameter (k)	4
Arrival rate (λ)	1.8
Mean time to complete service ($1/\mu$)	0.416667
Test time (t)	6.

Plot Parameters: (Clear contents to turn off chart)

Maximum time for $Wq(t)$ plot	5.
-------------------------------	----

Solve

Results:

Mean interarrival time ($1/\lambda$)	0.555556
Service rate (μ)	2.4
Fraction of time the server is busy (ρ)	75.00%
Fraction of time the server is idle (ρ_0)	0.25
Expected system size (L)	2.15625
Expected queue size (L_q)	1.40625
Expected waiting time in the system (W)	1.197917
Expected waiting time in the queue (W_q)	0.78125
Probability of no wait (q_0)	0.25
Probability that delay > t (PTW_q)	0.001793

Appendix (2) – f

Output for bulk models in QTS PLUS

1) Unlimited capacity, fixed-size bulk input Poisson queue in QTS PLUS

Input Parameters:

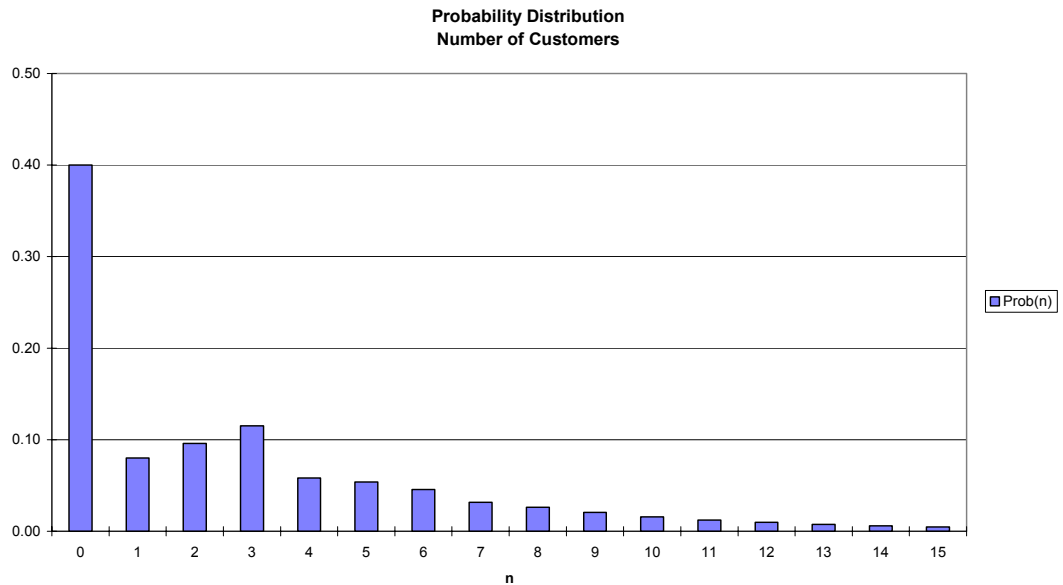
Arrival rate(λ)	1.
Mean service time ($1/\mu$)	0.2
Fixed batch size ($K > 1$)	3

Plot Parameter:

Maximum size for probability chart	15
------------------------------------	----

Results:

Mean interarrival time ($1/\lambda$)	1
Service rate (μ)	5
Server utilization (ρ)	60.00%
Probability of empty system (p_0)	0.400000
Mean number of customers in the system (L)	3
Mean number of customers in the queue (L_q)	2.4
Mean wait time (W)	1
Mean wait time in the queue (W_q)	0.8



Appendix (2) – f-contd

2) Batch Erlang input to single exponential server in QTS PLUS

Input

Parameters:

Arrival rate (λ)	1.8
Interarrival Erlang shape parameter (j)	2
Mean time to complete service ($1/\mu$)	0.416666667
Input batch size (K)	4

Results:

Mean interarrival time ($1/\lambda$)	0.555555556
Service rate (μ)	2.4
Fraction of time the server is busy (ρ)	75.00%
Fraction of time the server is idle (p_0)	0.25000
Expected system size (L)	4.03827978
Expected queue size (Lq)	2.278709835
Expected waiting time in the system (W)	1.682616575
Expected waiting time in the queue (Wq)	1.265949909

Roots:

-7.96492288055823E-002+0.487921487703715i
-4.53E-01
-7.96492288055807E-002-0.487921487703715i
0.827934423

Appendix (2) – g

Single server Markovian queue with two priority classes in QTS PLUS

One service rate and nonpreemptive, with lower number having higher priority.
[NOTE: The combined state probabilities are the same as for M/M/1.]

Input Parameters:

Mean arrival rate for class 1 (λ_1)	0.65
Mean arrival rate for class 2 (λ_2)	1.2
Mean service time ($1/\mu$)	0.333333

Plot Parameter:

Maximum size for probability chart	10
------------------------------------	----

Results:

Overall arrival rate (λ)	1.85
Mean interarrival time ($1/\lambda$)	0.540540541
Service rate (μ)	3
Server utilization (ρ)	61.67%
Probability of an empty system (p_0)	0.383333
Mean number of customers in the system (L)	1.608695652
Mean number of customers in the queue (Lq)	0.992028986
Mean wait time (W)	0.869565217
Mean wait time in the queue (Wq)	0.536231884

Priority Class #1:

Expected time in the system (W1)	0.595744681
Expected waiting time in the queue (Wq1)	0.262411348
Expected number in the system (L1)	0.387234043
Expected number in the queue (Lq1)	0.170567376

Priority Class #2:

Expected time in the system (W2)	1.017884675
Expected waiting time in the queue (Wq2)	0.684551341
Expected number in the system (L2)	1.22146161
Expected number in the queue (Lq2)	0.82146161

Appendix (2) – h

M/M/c queue in series in QTS PLUS

To setup new problem, enter number of nodes in queueing network.
Enter external arrival rate and upper bound on probability range,
then press "Solve".

Number of Nodes: 3
External arrival rate: 5.
Maxium no. of probabilities: 5

Solve

Node	$1/\mu$	Servers
1	0.2	4
2	0.4	5
3	0.3	3

Results

Interarrival time ($1/\lambda$)	0.2
Expected number in the system (L)	4.783445821
Expected number in all the queues (Lq)	0.283445821
Expected time to clear the system (W)	0.956689164
Expected delay in all the queues (Wq)	0.056689164
Probability that all servers are idle (p0)	0.010388444

Node Performance Measures

Node	1	2	3
Servers	4	5	3
μ	5	2.5	3.333333333
r	1	2	1.5
ρ	25.00%	40.00%	50.00%
p0	0.36735	0.13433	0.21053
L	1.006802721	2.039800995	1.736842105
Lq	0.006802721	0.039800995	0.236842105
W	0.201360544	0.407960199	0.347368421
Wq	0.001360544	0.007960199	0.047368421

Marginal Distributions

0	0.367347	0.134328	0.210526
1	0.367347	0.268657	0.315789
2	0.183673	0.268657	0.236842
3	0.061224	0.179104	0.118421

Appendix (3)

Output for steady state behaviour in MC QUEUE

Parameters :c= 1 rho= 0.6

Results : Pwait = 0.85

Pbusy = 0.85

E(W_q) = 5.6667

E(L_q) = 4.8167

Std(W_q) = 6.591

Std(L_q) = 6.017

Accuracy Check = 1.57E-16

Appendix (4) – a

List of functions in QTP

QTPMMS_ Functions

QTPMMS_TrafficIntensity:	Average traffic intensity.
QTPMMS_Util	: Average server utilization.
QTPMMS_L	: Average number of customers in system.
QTPMMS_Lq	: Average number of customers in queue.
QTPMMS_W	: Average time in system.
QTPMMS_Wq	: Average time in queue.
QTPMMS_PrEmpty	: Probability that system is empty.
QTPMMS_PrWait	: Probability that a customer will have to wait.
QTPMMS_PrFull	: Probability that a the system is full (and arriving customers- -do not enter)
QTPMMS_PrState	: Probability of observing the system in a particular state.
QTPMMS_ServiceLevel	: Probability that a customer will wait less than some- -threshol time)
QTPMMS_MinAgents	: Minimum number of servers needed to achieve a specified- -service level, assuming that the system capacity is fixed.
QTPMMS_MinServers	: Minimum number of servers needed to achieve a specified- -service level, assuming that queue capacity is fixed.

Appendix (4) – a-contd.

QTPGGS_Functions

QTPGGS_TrafficIntensity : Average traffic intensity.

QTPGGS_Util : Average server utilization.

QTPGGS_L : Average number of customers in system.

QTPGGS_Lq : Average number of customers in queue.

QTPGGS_W : Average time in system.

QTPGGS_Wq : Average time in queue.

QTPGGS_PrWait : Probability that a customer will have to wait.

Functions Provided for Backward Compatibility

QTPMMS_Po: Same as QTPMMS_PrEmpty, i.e., probability that system is empty. Note: the "o" is an "oh," not a zero.

QTPMMS_PrBalk: Same as QTPMMS_PrFull, i.e., the probability that the system is full (and arriving customers do not enter).

QTPMMS_Version: Same as QTP_Version, i.e., returns the version number for the Queueing ToolPak.

Other Functions

QTP_NORMSDIST: Calculates the cumulative distribution function for a standard normal random variable.

Appendix (4) – b

Sample output for model M/M/1 in QTP

arrival rate **8.99000000000** per hour
 service rate 9.00000000000 per hour
 servers **1.00000000000**
 queue capacity **inf**

Utilization 0.99888888889
 traffic intensity 0.99888888889
 average # of cars in the queue 898.00111111115
 average # of cars in the system 899.00000000004
 average time in the queue 99.88888888889
 average time in the system 100.00000000000

prob.of empty system 0.00111111111
 prob.of having to wait 0.99888888889
 prob. Of full system 0.00000000000

number of customers	probability
0	0.00111111111
1	0.00110987654
2	0.00110864335
3	0.00110741152
4	0.00110618106
5	0.00110495197
6	0.00110372425
7	0.00110249789
8	0.00110127289
9	0.00110004925

