Vol. 11, No. 3, May 2011, pp. 111–122
ISSN 1532-0545 | 11 | 1103 | 0111

INFORMS
Transactions on Education

DOI 10.1287/ited.1110.0064
© 2011 INFORMS

Teaching Note—Fitting a Theoretical Model to a Real Queue

Don McNickle
Management Department, University of Canterbury, Christchurch 8140, New Zealand, don.mcnickle@canterbury.ac.nz

A carefully structured practical queue-modelling assignment improves the understanding of queueing theory and teaches modelling and data-analysis skills. The assignment also demonstrates that it may be better to use models to estimate operating characteristics such as mean waiting times, even in circumstances where the system in question and the characteristics can be directly observed.

Key words: teaching queueing; data analysis; teaching statistics; formative assessment
History: Received: July 2010; accepted: January 2011.

1. Introduction

For most operations research (O.R.) techniques, a lecturer cannot expect students to carry out a complete application of the technique, from gathering the data to fitting the model. For many techniques, accessible examples where the students can also collect the data themselves simply do not exist.

Fortunately for queueing theory, examples of queues are all around us. I gratefully seize this opportunity to demonstrate the relevance of what I teach. For the last 20 years, I have had students in a 300-level O.R. class find an actual queue, collect data from it, and use the data to fit a theoretical model. They compare values of the measured queue characteristics with those predicted by the most appropriate model that they can manage. In spite of the students' limited range of models, the small amount of data they have time to collect, and their very limited knowledge of statistics and probability, the results of the assignment are often surprisingly good. The assignment exposes, and I hope helps correct, a gap in our business students' education when it comes to practical data gathering and analysis.

The ubiquity and accessibility of queues make them important and attractive models in the O.R. curriculum. In this journal, Mandelbaum and Zeltyn (2010) have described the development of a service engineering course largely based on queues, which features extensive data analysis. Ingolfsson and Grossman (2002) used spreadsheet simulations and graphs to demonstrate emergent properties of queues, also based on real data. The principal contribution of this teaching note is to show how the simple assignment described here is set up to have a strong formative component (Wikipedia 2011). Students are given a prescribed set of steps to follow and programs to use, and the assignment is structured so that along the way there are some checks on the calculations (§7) to ensure that the learning outcomes (§3) are achieved. The conventional approach when fitting a model to any kind of observable system would be to validate it by showing that the fitted model produces similar results (e.g., mean waiting times in this case) to those directly observed from an independent sample. As we shall see in §5, this method does not actually work here, or at least is very unreliable. Therefore, I must rely on other comparisons, calculations and demonstrations to achieve some of the learning outcomes. The reason why this validation method does not work, discussed in §5, shows students that even modest models have advantages, possibly even over direct observation of some characteristics.

2. The Students

The students, in a business school, have had two statistics courses but nothing previously on applied probability. We cover Markovian models, Erlang distributions (phase- or stage-type models), and the Pollaczek-Khinchin (P-K) formula for $M/G/1$ queues (Gross et al. 2008, §5.1) in this course. In preparation for this assignment, we cover the basics of stochastic processes, renewal and Poisson processes, and fitting distributions to data, in three lectures plus a tutorial. Almost all the material in §§3–6 of this note is distributed to the class before the assignment, either as part of the course reader for lectures (§§5, 7, and the appendix), the tutorial (§6), or in the instructions for the assignment (§§3 and 4). By structuring the problem, prescribing the steps, and providing suitable programs, I find that I can expect almost all of them to successfully fit the parts of a queueing model and practically explore the relationships between time and state processes.

3. Organization of the Assignment

In the instructions, students are required to

> Observe an actual queue. Collect enough data to fit a theoretical queueing model to your situation. Analyse the data following the steps described in lectures. Estimate the interarrival time and service time distributions from your data.
>
> Fit what you feel to be the most appropriate theoretical model (or models) of those we have studied.
>
> Measure the operating characteristics: the mean waiting time, the mean queue length, the state- and waiting-time distributions, by direct calculation from the data set and compare them with the theoretical values from your model. Reflect on the agreement between your observed and theoretical results.

Because the assignment does involve a number of steps, I allow students to work in pairs if they wish. Most do. They must seek permission from the relevant organisation to measure any queue that is not in a public space. Banks, especially, can be particularly sensitive about strangers observing them, and are probably best avoided. Students hand in the assignment electronically, including data files and analyses, so I can check their analyses if necessary. If the data have been recorded on paper, I also ask to see the original data sheets.

The two spreadsheets described in §5 are made available as examples for students to copy or modify. A set of spreadsheets for calculating steady-state queueing formulas, similar to those bundled with most O.R. texts, and the program mek1.m (a Matlab implementation of a simple direct approach to finding the state- and waiting-time distributions for $M/E_k/1$ queues) are available for them to use. The derivation of the formulas for mek1.m (in the appendix) is covered when we reach phase-type models by extending the derivation of the waiting-time distribution for the $M/M/1$ queue. Matlab may seem a strange choice for a business school, but most of my students have already encountered it in math courses. For those who have not, it only takes a brief demonstration to give them enough skills to get results out of mek1.m. I also use Matlab to teach Markov chains, where I find that its matrix orientation makes it very effective.

Four learning outcomes can be identified:

- the ability to measure and record real data;
- the ability to fit particular stochastic processes to real data;
- the ability to distinguish between time and event averages, to empirically validate Little's formula and other formulas for queueing characteristics, and to plot state- and waiting-time distributions;
- An appreciation of the value of even modest models in predicting emergent behaviour.

The average reported time to complete the assignment is 16 hours.

4. Selecting the Queue

Some general advice in the assignment relates the assumptions in the theoretical models that we have studied to the properties of appropriate queues:

- The situation should be unconstrained—no balking, reneging, abandonment, retrials, or state-dependent behaviour.
- The situation must have a clearly defined queue, with clearly defined service times, and each server must serve and complete only one customer at a time.

Most fast-food stores, for example, do not satisfy this assumption because the food is ordered and collected in separate actions. Generally, "people" queues can be a problem if there is complex customer behaviour. Supermarkets with several checkouts may not work well. Queue-selection decisions are complex, lanes open and close, and baggers move from lane to lane so that the service rate changes. Cars are often better behaved. My students find the queue of vehicles leaving a car park where customers pay on exiting is often a reliable choice. Queues at stop signs can work. Coffee shops, gas stations, and library issue desks are other possibilities. Single-queue multiple-server models can work well, even though the only practical models students have are $M/M/C$ or $M/M/C/N$, and the data-gathering is more complex. However almost all the students choose a single-server queue, so I will assume this from now on. Some other points of advice are:

- All of our models assume Poisson arrivals. At the least we must have renewal arrival processes, so avoid systems where arrivals can be bunched—such as road traffic downstream from traffic lights or stop signs.
- Pick a time when it is reasonable to assume that the arrival rate is constant. Modelling a café from 11 A.M. to 1 P.M. is usually not a good idea, because it is pretty much guaranteed to break this rule with different arrival rates from 11 A.M. to 12, and 12 to 1 P.M. Because queueing characteristics are nonlinear (often hyperbolic in the traffic intensity), modelling with averaged parameters does not give the average answer.
- The arrival process must be independent of the state of the system (fixed finite capacity is the one possible exception) and of the service-time process.

Examples that break this rule include ATMs in busy shopping malls, which may be almost always busy but have little queueing (state-dependent arrivals), and drive-in fast-food stores where there is little space between the ordering and food-collection windows (blocking, balking, and non-Poisson arrivals at the collection window).

- A traffic intensity of about $\rho = 0.5$ is good, giving both some queueing and easy data collection.

5. Collecting the Data

Two questions immediately arise: What data should I actually collect? How much data do I need? The first question is almost always "How many observations do I need?" As usual, the answer is "How small a confidence interval for the mean do you want?" For a negative exponential distribution, I can expand this answer because the mean and standard deviation are equal. Therefore, if I want a 95% confidence interval for the mean to be (say) 10% of the mean (a relative precision of 10%), then $0.1 \approx 1.96/\sqrt{N}$, or $N = 384$. As a result, we start the discussion from this number. This sample size comes as a shock to most students, and we usually compromise on at least 100 observations, but they are required to calculate confidence intervals for the means of the service and interarrival times and make some assessment of the errors such small samples will produce. Often this sample size of about 100 is in fact dictated by the essential requirement to limit data collection to a period during which the arrival rate is constant (384 arrivals take 6.4 hours if they arrive at one per minute.)

Now is a good time to introduce the fact that for serially correlated quantities like queue lengths or waiting times, the confidence intervals are orders of magnitude larger. From the formulas in Daley (1968), the expected number of customers needed to give estimated mean waiting times with relative precisions of 10% or 5% in some simple queueing models can be calculated with the aid of Maple, as shown in Table 1.

Table 1 Expected Number of Observations to Reach Relative Precisions of 10% and 5% for the Mean Waiting Time

Traffic	$M/E_6/1$		$M/E_3/1$		$M/M/1$	
intensity	10 (%)	5 (%)	10 (%)	5 (%)	10 (%)	5 (%)
0.1	7,768	31,072	8,723	34,890	11,671	46,687
0.2	5,230	20,919	5,975	23,903	8,547	34,190
0.3	4,805	19,220	5,551	22,204	8,276	33,105
0.4	5,140	20,561	5,975	23,903	9,134	36,537
0.5	6,170	24,684	7,192	28,769	11,140	44,562
0.6	8,334	33,337	9,710	38,842	15,110	60,441
0.7	13,121	52,485	15,249	60,998	23,677	94,710
0.8	26,604	106,418	30,786	123,144	47,443	189,775
0.9	97,166	388,664	111,797	447,190	170,268	681,072

Therefore, if you want to measure the average waiting time in an $M/M/1$ queue with a traffic intensity of 0.9 to an accuracy of about ±5%, you will need to observe about 681,072 customers, and the reason for this is mostly because the waiting times are highly correlated. If I have waited 20 minutes in a queue, there is a good chance the person behind me will also wait for a similar amount of time—in fact, the correlation between my and their waiting time in this situation is 0.99043 (Daley 1968). Observing 681,072 customers arriving at (say) one-minute intervals, takes nearly four years if you work an eight-hour day! Even for the quite likely situation of an $M/E_3/1$ queue with a traffic intensity of 0.5, 5% accuracy will require 28,769 observations—two months' work. This is starting to suggest that even if direct measurement of characteristics like waiting times is possible, it may not be practical.

Another way of characterising this effect of correlation is to show how bad the results are from direct measurement on smaller samples. Figure 1 shows the result of simulations of what happens if 1,000 observers were each to estimate the mean waiting time by collecting samples of 100 successive waiting times from an $M/E_3/1$ queue with a traffic intensity of 0.5.

The wide distribution of these estimated values raises three interesting points. First, it shows it is difficult to prove that any particular queueing model is right or wrong by direct observation of the queue characteristics. Validation of the model can practically only be done by showing that all of its parts (arrival processes, service times, queue disciplines) are validly modelled. Second, along with Table 1, it shows that unless sample sizes are exceptionally large, a confidence interval for the mean waiting time will be so wide as to be useless (in addition, it shows that the distribution of the average waiting times from samples of 100 is still distinctly asymmetric; hence,

Figure 1 The Results of Directly Measuring Average Waiting Times in an $M/E_3/1$ Queue from Samples of 100 Waiting Times

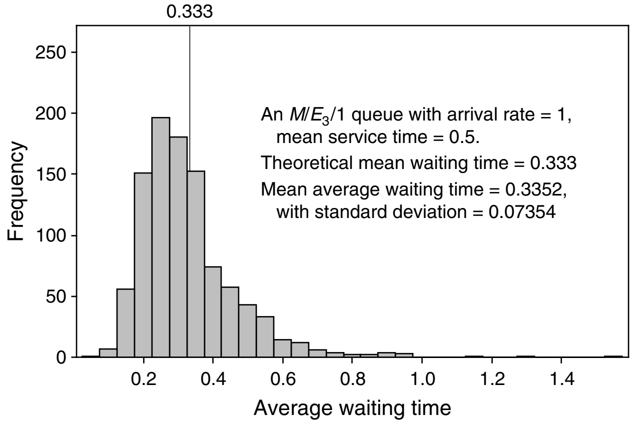
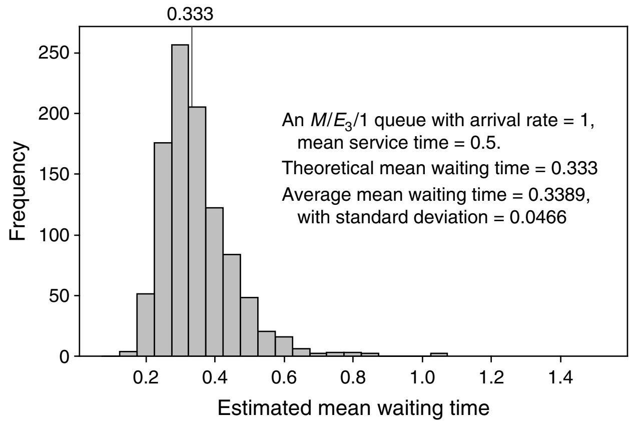


Figure 2 **Estimating the Arrival Rate and Mean Service Time from Samples of 100, Then Calculating Mean Waiting Times from the P-K Formula**



a confidence interval will be difficult to calculate.) Therefore, the conventional approach of validating the model by showing that it produces similar results to those directly observed from an independent sample of observations does not actually work here—or at least is very unreliable.

A third point for discussion is that the uncertainty in the results raises the possibility that if our queue matches a theoretical model reasonably well, we will do better by fitting the arrival and service distributions and using the theoretical formula, than we would by measuring the average queue length or waiting time directly. Figure 2 shows the histogram we get if the same 1,000 observers were to spend their time estimating the arrival rate and mean service time from samples of 100 observations and then used the P-K formula to estimate the mean waiting time. The histogram of estimated mean waiting times is much narrower than that of the observed average waiting times in Figure 1. Therefore, some good points can be made for the value of using even modest models here.

The second question is "What data do I collect?" Students should work out what they need to collect before they start the assignment and design a form for collecting it. The events in a queue are arrivals, entry to service, and departures, so collecting the times of these is one possibility. However, for the purposes of this analysis, if customers are waiting, the departure time of one customer must correspond to the entry to service time of the next, i.e., the times for any dawdling at the service facility, time to move up from the queue to the server,[1] servers who talk to colleagues, must be included in the "service times." Therefore, for single-server queues, recording arrival and departure times is both enough, and evades this source of error.

If A_n, E_n, and D_n are the times of arrival, entry to service, and departure for the nth customer, then all the needed times can be found from:

1. the interarrival time between the $n-1$th and nth customer is $A_n - A_{n-1}$;
2. $E_n = \max(A_n, D_{n-1})$;
3. the service time of nth customer is $D_n - E_n$; and
4. the waiting time of the nth customer is $E_n - A_n$.

I encourage collection of the waiting-time data from the same sample as will be used to estimate the parameters. The uncertainty in the sampled average waiting time is already large enough for the reasons given above. As discussed previously, Table 1 and Figure 1 show that validating the model from the results of a small independent sample is unlikely to work, whereas getting the queue characteristics from the same sample allows exploration of how the characteristics are made up from the data, and also gives some handy checks on the calculations later on. Better students do raise the possibility that they should return to the system the next day to get a small independent sample of waiting times for validation. I of course support the idea in principle, but warn them that as Figure 1 shows, the results of this are often not very convincing.

Simultaneously recording the state (number of customers in the system) at arrival and departure times would be desirable, because we will need these data to estimate the state distribution. This is, however, often difficult and prone to errors if the system is moving quickly. If they have not been collected, state data can be reconstructed as follows. For general queueing models the state distribution is estimated by the fractions of time spent in particular states, and L and L_q (the estimated mean number of customers in the system, and in the queue) are time averages. Therefore, we need the data to be able to calculate these. Each arrival increases the state by 1, and each departure reduces it by 1, so:

1. Put a column of 1s next to the arrival times and -1s next to the departure times.
2. Merge the arrival and departure times into one column, carrying along the 1s and -1s into the adjacent column.
3. Sort the single column of arrival and departure times into increasing order by time, again carrying along the relevant 1 or -1.

[1] There are, of course, numerous queueing models to exactly analyse these situations, but they add too much complexity here. One exception to this that students have used to analyse the situation where a waiting customer takes an appreciable time to move up to the server, whereas a customer arriving to an idle server goes straight into service, is the variation on the P-K formula in Welch (1964), which nicely handles queues where customers have an appreciable move-up time.

4. Now do partial sums of the column of 1s and −1s. This is the state, and the difference between successive times in the column of arrival and departure times is the time spent in that state.

A spreadsheet that illustrates this process, and also shows one method of sorting out the times spent in particular states, is `Estimating State Distributions.xlsx`.[2]

If you have a package that can do plots in a "stair" or "step" style—Minitab or Matlab, for example—students may like to plot the state against time and check that the average height of this graph is L. This is not essential but is good to see.

Increasingly, students competent with Microsoft Excel write macros to record times directly into a spreadsheet. A simple one that records arrival and departure times in the hr:min:sec format is attached as `Time Recorder.xlsm`. Although this simplifies data recording, students have to be careful to ensure that no events are missed. Paper recording of arrival, entry to service time, and departure times is possibly safer, in that missed events are more likely to be spotted.

Although Excel is adequate for recording the data and calculating interarrival and service times, a proper statistical package makes data analysis and graphing a lot easier. Of the major packages, Minitab seems best at the tasks, although SPSS and SAS can do most of them. In Excel it pays to first change the cell format to General to convert hr:min:sec data to fractions of a 24-hour clock and to further convert this to, say, decimal minutes. This gives numbers that are more intuitive and that are also safer to use in formulas or to cut and paste between the packages. An example of what can happen when formats are mixed is that if hr:min:sec data is squared in Excel, say to calculate the variance of the service times for the P-K formula, Excel automatically converts it to units of fractions of a 24-hour-clock squared, so errors calculating the P-K formula are possible here. The discussion of Figure 5 in §6.1 also shows how the small size of times measured in 24-hour-clock units has the potential to mislead.

Students must do some analysis promptly after they collect their data if they want to avoid unpleasant surprises later. This should include histograms, and a table of means, standard deviations, and coefficients of variation (CV), for both the interarrival and service times. For there to be any chance of a Poisson arrival process, we need a CV close to 1. An idea of how close "close to 1" is can be found by using the fact that the reciprocal of the CV is a scaled and shifted version of the statistic used in the t-distribution. Therefore, for a process with a CV of 1, 95% of the CVs estimated from samples of N observations should fall in an interval $\{1/(1+1.96/\sqrt{N}), 1/(1-1.96/\sqrt{N})\}$. For $N = 100$ this interval is $\{0.836, 1.244\}$. Estimated arrival CVs outside this range should cause closer examination of the data and the situation. At this point they should also draw the plots for checking stationarity and independence (Figures 4–6 in §§6.1 and 6.2) to guard against being caught later by some of the data problems that these figures may show up. If there are serious problems, the best advice at this point may be to either re-collect the data or find another system, to guarantee that the formative aims of the assignment are achieved.

6. Fitting the Parts of the Model

Good references for fitting and estimation from which much of this section is adapted are the early editions of Gross and Harris (1st edition 1974, §7.1, 2nd edition 1985, §6.6), Hall (1991, Chapters 2 and 3), and Law (2007). Discrete-event simulation texts often have similar material because the data analysis and distribution-fitting requirements of queueing and simulation are similar. Monte Carlo and some discrete-event simulation software come with automated distribution fitters such as ExpertFit and Stat:Fit. These do a good job of distribution fitting, but may recommend distributions that are too complex to be of value, or that allow negative interarrival or service times.

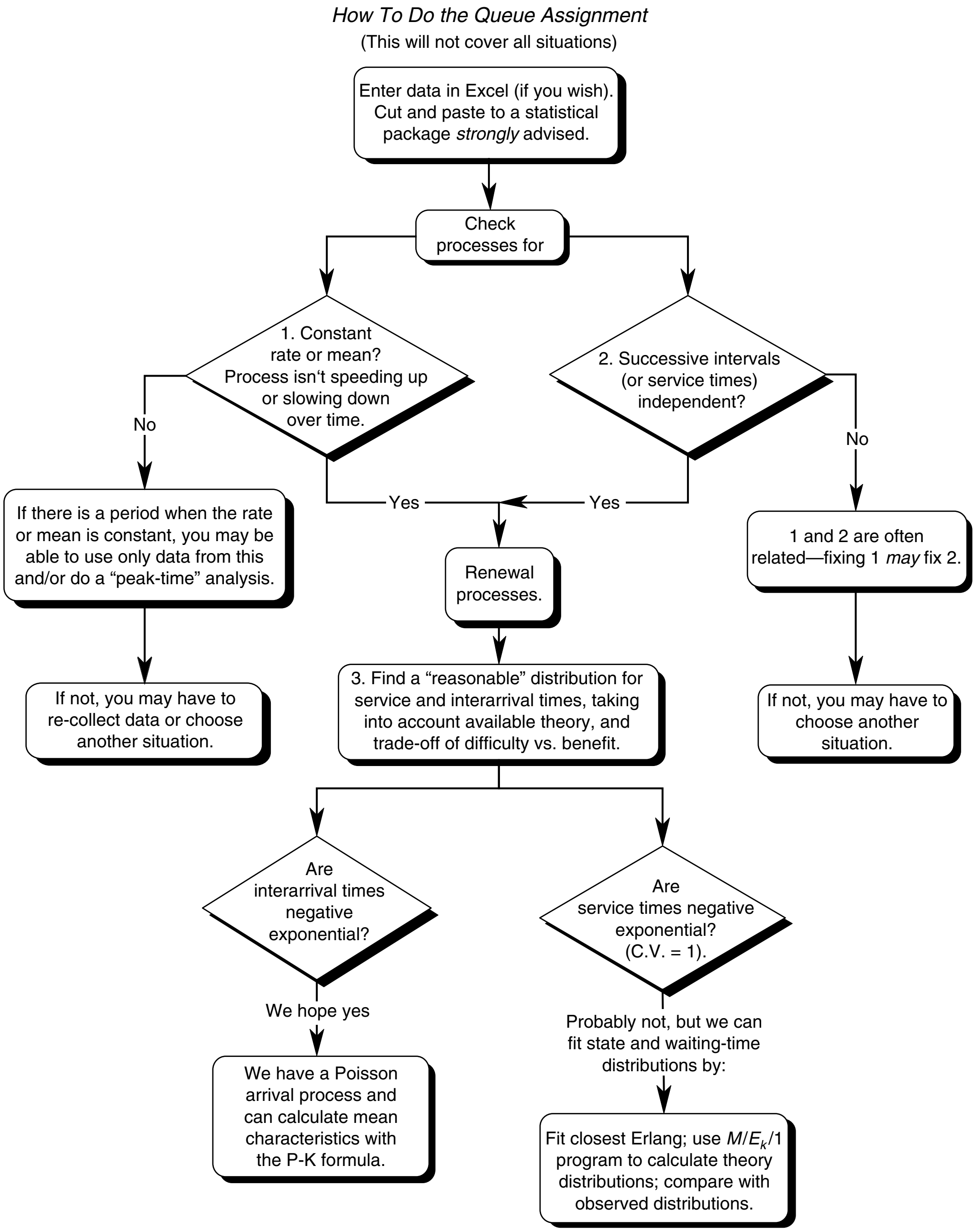
Simplicity and automation are essential here if the assignment is not to become too difficult.[3] I teach estimation by moments, graphical analysis, and the use of whatever tests are included in the available statistical packages. Students estimate means (and variances) only, and I avoid the mention of rates during the data analysis phase because they can cause confusion, and in any case students do not yet know that they have any Poisson processes.

Because this is both a summative and formative assignment, I give an explicit set of analysis steps for students to follow, summarised in the flow chart in Figure 3, and then work with them on any specific problems that these throw up. For example, if a student finds that their service times have a hyperexponential distribution, then I first get them to complete the assignment with the best model they have—$M/M/1$ in this case. Then I calculate the waiting time and state distributions for an appropriate $M/H_2/1$ model, and give them the formulas for these to improve their answer.

[2] This and the other two supplemental files are available from http://ite.pubs.informs.org/.

[3] Much has been published on statistical analysis for queues. See, for example, Bhat et al. (1997). However, the results are often very specific to particular theoretical models, and are often difficult to explain or implement at this level.

Figure 3 **A Flow Chart for the Modelling Process**


How To Do the Queue Assignment
(This will not cover all situations)
Enter data in Excel (if you wish). Cut and paste to a statistical package *strongly* advised.
Check processes for
1. Constant rate or mean? Process isn't speeding up or slowing down over time.
2. Successive intervals (or service times) independent?
No
If there is a period when the rate or mean is constant, you may be able to use only data from this and/or do a "peak-time" analysis.
If not, you may have to re-collect data or choose another situation.
Yes
Yes
Renewal processes.
No
1 and 2 are often related—fixing 1 *may* fix 2.
If not, you may have to choose another situation.
3. Find a "reasonable" distribution for service and interarrival times, taking into account available theory, and trade-off of difficulty vs. benefit.
Are interarrival times negative exponential?
We hope yes
We have a Poisson arrival process and can calculate mean characteristics with the P-K formula.
Are service times negative exponential? (C.V. = 1).
Probably not, but we can fit state and waiting-time distributions by:
Fit closest Erlang; use $M/E_k/1$ program to calculate theory distributions; compare with observed distributions.

Students need to check these three basic assumptions:

1. The processes are stationary: i.e., the mean time between events is constant as time passes. Arrival processes, in particular, often do not have this property.

2. Successive interarrival or service times are independent (uncorrelated).

3. Interarrival and service times are compatible with particular probability distributions.

If they get positive answers to 1 and 2, their results are compatible with a renewal process, and if in addition the "particular distribution" in 3 is negative exponential, they have a Poisson process.

6.1. Testing for Stationarity

Following Hall (1991), we plot the arrival number (vertical axis) against the arrival time (horizontal axis) to check that the arrivals satisfy Assumption 1.

What we are looking for is that the slope of this graph (the arrival rate) does not change over time.[4]

[4] All the graphs from here on are taken, with permission, from actual assignments. They have been left as the students submitted them, aside from some minor editing of titles.

Figure 4 An Unsatisfactory Plot of Arrival Number Against Arrival Time

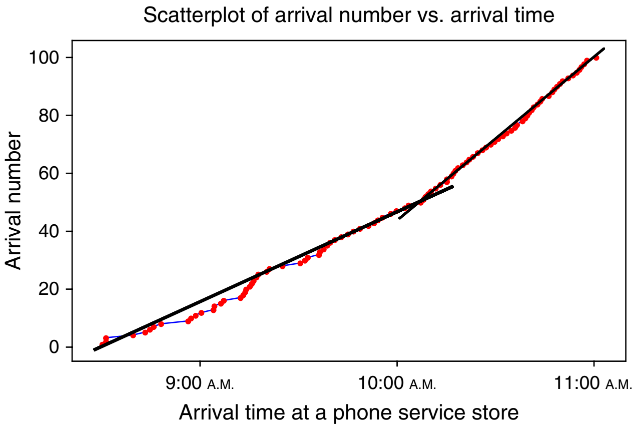


Figure 4 shows a considerable deviation from this assumption. Up to 10 A.M., the arrival rate (at a mobile phone service store at Christchurch Airport) is 0.51 customers per minute. After 10 A.M. it suddenly increases to about 0.78 customers per minute, probably due to the arrival of an international flight. This change could be confirmed statistically by estimating the two mean interarrival times separately and using the usual t-test for differences in means of two random samples. If unnoticed, this mixing of two or more (possibly exponential) distributions with different rates often manifests itself as an apparent hyperexponential distribution for interarrival times.

This is not good news. Because the formulas for queueing characteristics are nonlinear, using average parameters will not produce average answers. Fortunately in this case, the student had enough data to fit the model to the period from 10 A.M. onward.

Service times can be checked to see that there is no gross change over time by plotting service times against customer number, and checking the significance of the slope of the trend line (Figure 5). Minitab can fit a regression line automatically to this scatter plot, but the significance needs to be checked by doing a separate regression. The small values for the service times, and hence the apparently zero slope of this graph, are because the student has used fractions of a 24-hour clock as units. These small values can easily disguise a significant slope unless the test is done. A sudden change in mean service times can be tested by a t-test as for interarrival times.

Figure 5 Checking for Trends in Service Times

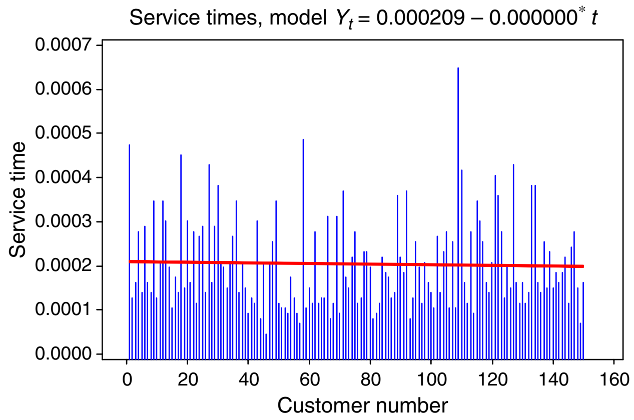
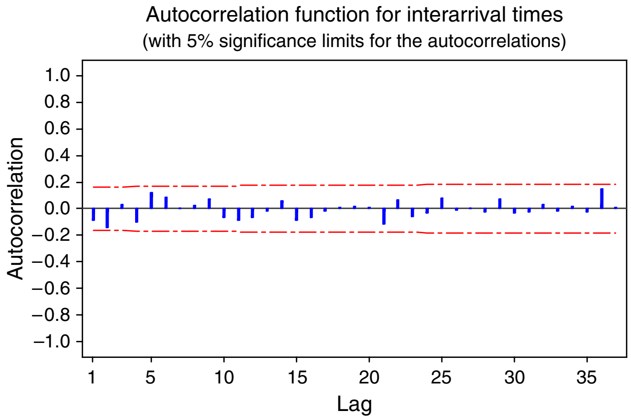


Figure 6 Checking for Independence with a Plot of Autocorrelations



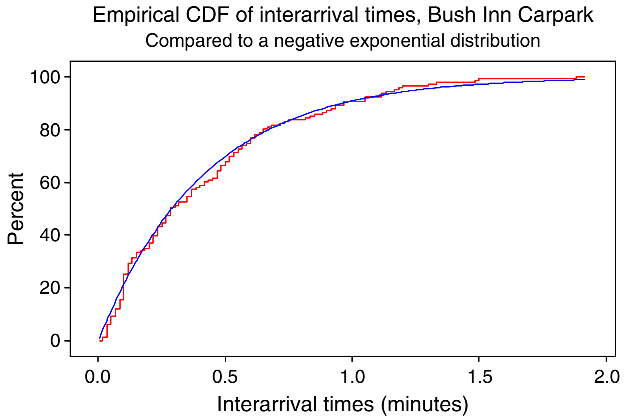
6.2. Testing for Independence

A modest but easy check for correlation in service or interarrival times can be done by using the time series section of the package to plot the autocorrelations for the first few lags. This can be explained to students who have not done a forecasting course as (approximately) the ordinary correlation of the series with previous (lagged) values, plotted for a number of lags. Figure 6 shows a satisfactory result, with all or almost all of the autocorrelations within the acceptance region for the null hypothesis of no serial correlation. I find one of the most common reasons for autocorrelation in queues is state-dependent arrival or service processes. (An ATM in a mall is an example that may show both properties. Customers lurk nearby, looking at the shops until the machine is idle, or they punch the buttons faster if others are waiting. A symptom of this is that the machine is almost never idle but also never has a large queue.) Modelling these attributes would mean estimating several Poisson rates or service distributions instead of one, so the data requirements increase significantly.

6.3. Testing for Assumption 3: Particular Distributions

Many O.R. texts still recommend a chi-square test on the counting (Poisson) distribution for testing for a Poisson process. Because we have measured times of events rather than counting the number of events in an interval, this is not an intuitive idea. Also, a Poisson process is one of the few stochastic processes that have a simple counting distribution, so the method cannot be extended to other processes. A more direct

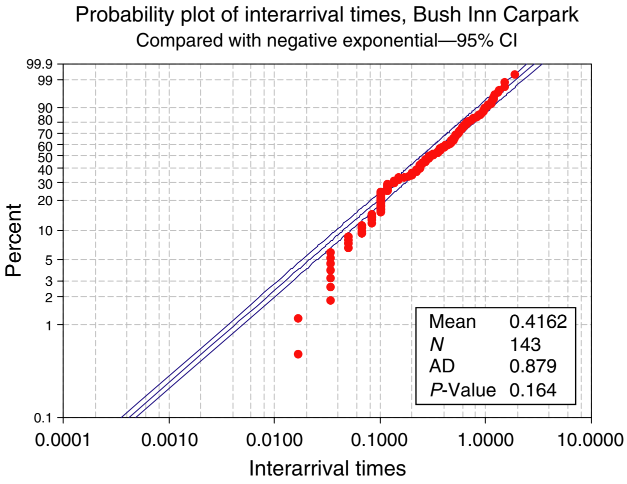
Figure 7 **Comparing the Sample cdf of Interarrival Times with That of a Negative Exponential Distribution with the Same Mean**



approach is simply to see if the sample cumulative distribution function (cdf) has the right shape. Does it look like the proposed theoretical distribution function? If nothing else, the sample and theoretical cdfs can be just be plotted on the same graph and compared by eye. This can be done with Excel. However Minitab has a nice "Empirical CDF" function in the graph menu, which automatically draws this graph for a wide range of distributions.

Even without Minitab, the Kolmogorov-Smirnov test statistic for distributions (the maximum vertical deviation of the sample cdf from the theoretical cdf) is simple enough to calculate directly in Excel. There are more sophisticated test statistics based on the same idea. For example, the Anderson Darling test statistic is a weighted sum of all these deviations. In Minitab this is available in "Probability Plot," also down the graph menu. Thus, from Figure 8 the Anderson Darling ("AD") test statistic for the Figure 7 data is 0.879, and from the P-value of 0.164 we would accept the null hypothesis that these data came from a negative exponential distribution at the 16% level of significance. The clustering of points at small interarrival times is caused by the data being recorded only to the nearest second and does not have much significance if the sample is large enough.

Figure 8 **The Minitab P-P Plot for the Data in Figure 7**



6.4. Fitting Service-Time Distributions

At this point you can choose to finish the assignment. If the arrival process is reasonably Poisson, then the observed average waiting time in the queue, W_q, can be compared with the theoretical answer from the P-K formula for an $M/G/1$ queue. And if the average number of customers in the queue, L_q, has been calculated from the state data the student can verify that $L_q \approx \lambda W_q$. This relationship should be almost exact if the observations ended with an empty queue.

However, I think the assignment works better and accomplishes much more if students can carry on to find theoretical state- and waiting-time distributions, and compare them with the observed distributions. Most of my students have never actually plotted distributions before, so simply doing this is a valuable exercise. As Table 1 and Figure 1 suggest, the theoretical and observed values of the mean waiting time and mean state often will not agree well, whereas the theoretical and observed waiting-time- and state distributions usually fit each other much better, admittedly caused in part by measuring the waiting times and state probabilities from the same sample of customers.

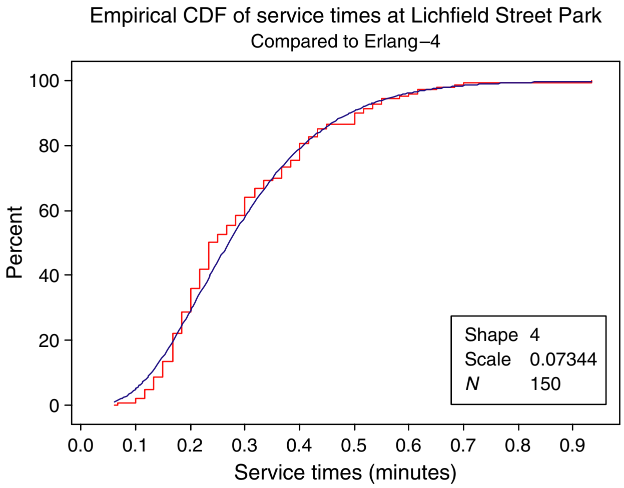
To find state- and waiting-time distributions, students must now fit a particular distribution to their service times. Service times for the systems we observe are usually less variable than negative exponential, so we need a fitting family of distributions with CVs < 1. The only practical family of distributions at this level are the Erlang-k distributions. The parameter k is taken as the closest value to $1/\mathrm{CV}^2$, and adjacent values of k can also be tested. Minitab's Empirical CDF function does not know about Erlang distributions, but it does know about Gamma distributions, and they are the same thing for integer values of the shape parameter k. Often, values of k in the range 4 to 9 give surprisingly good visual fits to the observed service time cdf, and results like those of Figure 9 are typical.

Frequently, however, the Anderson-Darling test statistic will be statistically significant, because unlike the negative exponential case there is usually no fundamental reason why an Erlang distribution should fit. For students who are stuck on this point, I go through these questions:

What is the chance you will actually do much better if we could find a better fit for the service-time distribution?

- The probability of zero customers in the system will still match that calculated from the mean service and interarrival times and hence is unchanged.

Figure 9 **Comparing Service Times with the Nearest Erlang Distribution**



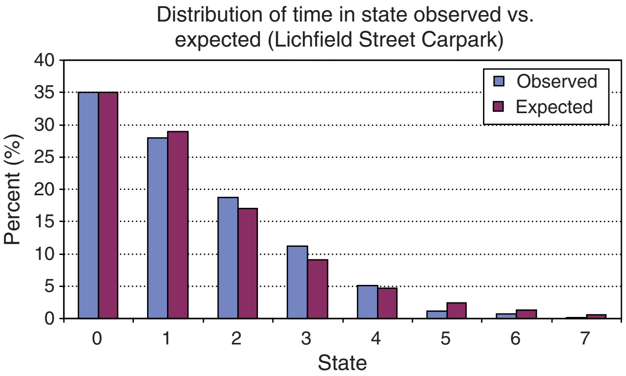
- If the arrival process is Poisson and we have managed to accurately approximate the variance of the service times, then from the P-K formula the mean number of customers in the queue is also correct.

Therefore, how likely is it that the rest of the state distribution can be too far off? Now consider the trade-off between model complexity and increased accuracy. What will the payoff be for the extra effort in fitting a more complex distribution and solving the resulting queueing model?

Very often this argument is accepted, and Figure 10 shows the comparison of the theoretical and observed state distributions from the assignment whose service times were used for Figure 9.

For this part of the assignment to work, students need an efficient, reasonably easy to understand method for calculating the state- and waiting-time distributions for an $M/E_k/1$ queue. Most texts give only a probability generating function for the state distribution and a Laplace-Stieltjes transform for the waiting-time distribution. Possibly as a result of this, most queueing packages give only the Pollaczek-Khinchine mean values for this model, so I provide a stand-alone program. The appendix and the Matlab file `mek1.m` give an implementation of a method that only relies on balance equations and probabilities to work and hence can be explained at this level. The basic theory behind this is covered when we discuss Markovian models and the balance-equation method.

Figure 10 **A Comparison of Observed and Theoretical State Distributions**



7. Results

When set up like this and carried through to this point, the assignment is largely self-checking, helping the students and saving me a lot of calculation and reanalysis when marking.

- The estimated probability of no customers in the system should be almost exactly that predicted by theory $-1-\rho$. It is just a rearrangement of the same data that went into calculating the service and interarrival time means.
- The fraction of customers who do not wait should be close to 1 - ρ, but need not be exact.
- Little's formula will relate the observed values of L_q and W_q almost exactly if the observations finished with the server idle.

Figure 10 shows a typical fit of the observed and theoretical state distributions. Estimating the state distribution and the parameters from the same data ensures that the theoretical and estimated probabilities that the system is empty have exactly the same value, and usually the other theoretical and estimated probabilities of the other states will not be too far off. Thus, the student gets a more rewarding result than that from comparing the theoretical and estimated values of L_q and W_q. Figure 11 from another assignment shows a comparison of the observed waiting-time distribution with that produced by `mek1.m`.

Finally, you may see empirical CDFs or other graphs as in Figure 12.

There appear to be no service times with lengths between 0.6 and 1, 1.6 and 2, and 2.6 and 3. This is the

Figure 11 **Observed and Theoretical Waiting-Time Distributions**

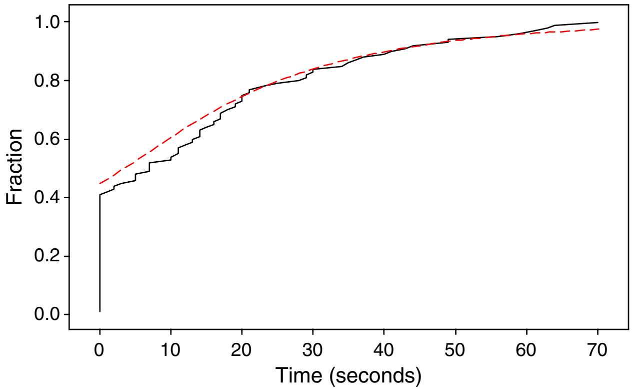
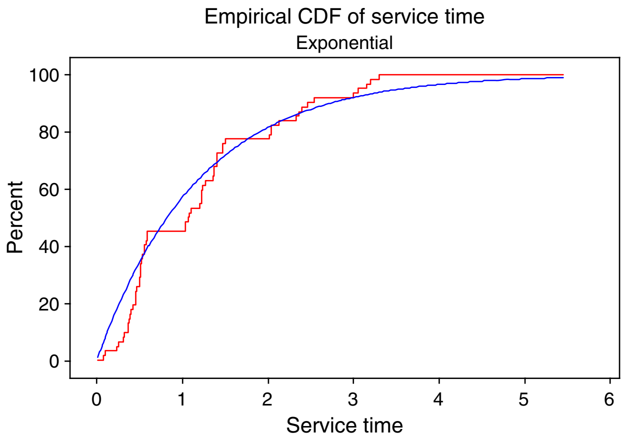


Figure 12 **Problems with Data Recording**



unfortunate consequence of students recording their data on paper in minutes and seconds, perhaps with a dot separating the minutes and seconds, and then returning to the data some time later (possibly the night before the assignment is due) and thinking it was recorded in decimal minutes! I get about one of these a year.

8. Outcomes and Conclusions

About 95% of the class do a good job. They select an appropriate model and correctly carry out the steps to fit it. About 85% find something like a Poisson arrival process. Usually the theoretical model gives a larger value for the mean queue length or waiting time than they observed. Over a sample of 60 assignments, the average discrepancy was 12%. Reasons for this bias probably include:

(i) we have estimated the mean waiting time over a fixed number of arrivals and then used the same sample to estimate the Poisson arrival rate, rather than counting the number of arrivals in an interval of fixed length;

(ii) with 100 observations the distribution of sample average waiting times is still very asymmetric, as Figure 1 suggests, so underestimation is more likely; and

(iii) students almost always start and finish their data gathering with an empty queue.

Although this biases the results, I do not discourage it since it means that the empirical versions of Little's formula work perfectly. Better students who raise this issue are encouraged to start their data collection from a preset (say, the 5th) arrival as a gesture towards solving this problem. Finally,

(iv) I think many of the situations students observe are actually finite capacity (but of unknown size). Servers speed up or customers don't join when the queue is large. But over small samples these effects are impossible to detect or measure.

Simulations have shown that biases (i), (ii), and (iii) almost disappear for sample sizes of 500 or more. So they are largely artefacts of the small sample sizes that are all we can afford to collect. I ask students to reflect on why their result is an over- or underestimate in their case, not forgetting the variability they can expect in the observed queue characteristics described in §5. We discuss these effects after the assignments are returned.

Could the assignment also be used for a graduate class? Yes, definitely. I have done this. If the students have come from an applied math or O.R. background with little experience of fitting models to data I believe they could greatly benefit from the assignment as it stands, presented as an easy exercise. It can be made a bit more challenging by not providing the spreadsheets or `mek1.m`, and further by extending the range of possible theoretical queueing models to whatever you have covered.

Does it work? In a review of the undergraduate O.R. degree this was the only assignment specifically mentioned by past students. Every year there are several positive comments included in the assignment answers, usually along the lines that this assignment made queueing models much clearer. My other assignments in this course do not attract similar comments! I see students using the methods they learn in this assignment to do assignments in other courses in operations management and quality control, suggesting that the first two learning outcomes are attained.

Supplementary Material

Files that accompany this paper can be found and downloaded from http://ite.pubs.informs.org/.

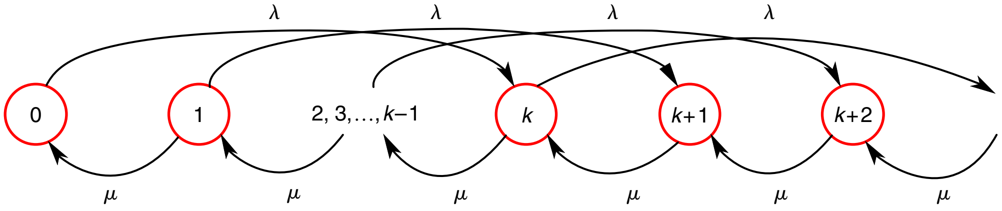
Acknowledgments

The associate editor and two referees provided careful comments that improved this paper considerably. Members of MSCI 302 2008 and 2009 agreed to the use of their data and graphs in this article. The original data recorder, `Time Recorder.xlsm`, was written by Chen Yi-an for his assignment. An earlier version of this assignment was also described in McNickle (1986).

Appendix A. Matlab Implementation of a Direct Method for Finding the State and Waiting-Time Distributions for An $M/E_k/1$ Queue

Normally, the textbook analysis of an $M/E_k/1$ queue ends up with a probability-generating function for the state distribution, or possibly an explicit form that involves either the roots of a polynomial of order k or potentially large binomial coefficients, and a Laplace transform of the waiting-time density. Neither is very satisfactory for explanation or calculation, especially to students with modest math skills. The extreme accuracy of packages such as Matlab or Maple, however, make an alternative possible, which once might have seemed like numerical suicide but now appears more accurate than using the analytical solutions. This method is certainly not original. Bits of it can

Figure A.1 **A Balance Diagram for $M/E_k/1$**



be found scattered through the queueing literature. For example, it is hinted at in Gross et al. (2008, p. 135), but not exploited there, although the state calculation method appears to be used in the latest version (3.1) of their software, QTSPlus. In a survey of 35 queueing books, I could only find the method described in books by Tijms (e.g., in a more general form in Tijms 2003a). When restricted to $M/E_k/1$, it can be satisfactorily explained at this level as one of the many extensions to the balance-equation method, and packaged into a reasonably user-friendly program in Matlab. This has been tested in class for two years, although a simpler version that does the state distribution only has been used for five years.

As usual, we start from the fact that an Erlang-k random variable has the same distribution as the sum of k negative exponentially distributed stages. I will use μ to be the rate of *one* of the stages, i.e., the mean service time is $s = k/\mu$. Therefore, the traffic intensity is $\rho = \lambda s = k\lambda/s$. Using this definition of μ rather than $s = 1/\mu$ simplifies the notation a bit. The steady-state balance diagram for the number of *stages* of service is in Figure A.1.

So if $\{p_n, n = 0, \ldots, \infty\}$ is the probability distribution for the number of exponential *stages* present in steady state, then

$$\begin{aligned}
\lambda p_0 &= \mu p_1, &&\text{or } p_1 = (\lambda/\mu)p_0\\
(\lambda+\mu)p_1 &= \mu p_2, &&\text{or } p_2 = ((\lambda+\mu)/\mu)p_1\\
&\vdots\\
(\lambda+\mu)p_{k-1} &= \mu p_k, &&\text{or } p_k = ((\lambda+\mu)/\mu)p_{k-1}\\
(\lambda+\mu)p_k &= \mu p_{k+1} + \lambda p_0, &&\text{or } p_{k+1} = ((\lambda+\mu)/\mu)p_k - (\lambda/\mu)p_0\\
(\lambda+\mu)p_{k+1} &= \mu p_{k+2} + \lambda p_1, &&\text{or } p_{k+2} = ((\lambda+\mu)/\mu)p_{k+1} - (\lambda/\mu)p_1\\
&\vdots
\end{aligned}$$

However, as for any conventional single-server queue, we know that $p_0 = 1 - \rho = 1 - \lambda s$ (zero customers is the same thing as zero stages of service), so the probabilities can all be calculated iteratively out as far as we need from the right-hand set of equations.

These *stage* probabilities are converted to *state* probabilities, of the number of customers in the system, by adding up appropriate terms. Call these $\{q_n, n = 0, \ldots, \infty\}$. So,

$$\begin{aligned}
q_0 &= p_0\\
q_1 &= p_1 + p_2 + \cdots + p_{k-1}\\
q_2 &= p_k + p_{k+1} + \cdots + p_{2k-1}\\
&\vdots
\end{aligned}$$

By conditioning on the number of stages of service that an arriving customer sees waiting to be served, and using the fact that an arriving customer sees the steady-state distribution of the number of stages,

$$P(w_q \le t) = p_0 + p_1 E_1(t) + p_2 E_2(t) + \cdots,$$

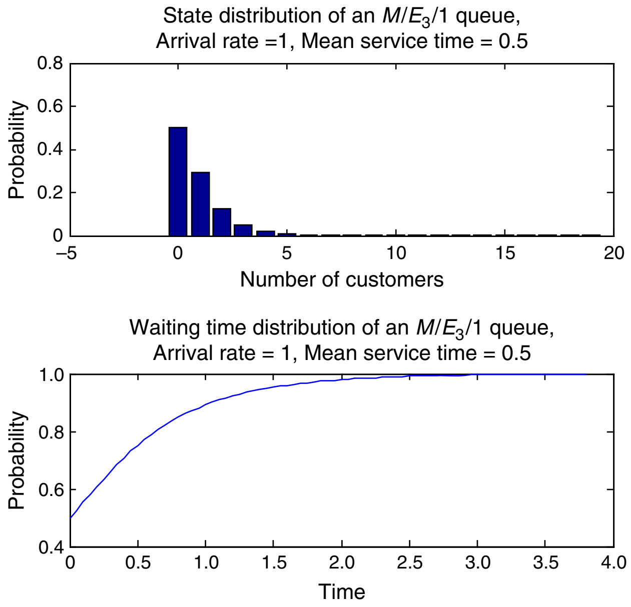
where w_q is the waiting time in the queue, and $E_r(t)$ is the cumulative distribution function for an Erlang-r distribution, so

$$\begin{aligned}
P(w_q \le t) &= p_0 + p_1(1 - e^{-\mu t}) + p_2\left(1 - \sum_{j=0}^{1} \frac{e^{-\mu t}(\mu t)^j}{j!}\right)\\
&\quad + p_3\left(1 - \sum_{j=0}^{2} \frac{e^{-\mu t}(\mu t)^j}{j!}\right) + \cdots.
\end{aligned}$$

This expression is exactly the same as one that can be used for deriving the waiting time in an $M/M/1$ queue (see (Gross et al. 2008, p. 65) for a slightly different form), except that here $\{p_n, n = 0, \ldots, \infty\}$ is the distribution of the number of stages rather than the state distribution. Therefore, in class I simply extend the $M/M/1$ waiting-time derivation. Rearranging and collecting terms gives

$$\begin{aligned}
P(w_q \le t) &= 1 - (1 - p_o)e^{-\mu t} - (1 - p_0 - p_1)e^{-\mu t}(\mu t)\\
&\quad - (1 - p_0 - p_1 - p_2)\frac{e^{-\mu t}(\mu t)^2}{2!} - \cdots.
\end{aligned}$$

Figure A.2 `mek1.m` **Output**



This is a simpler form of expression 5.5.6 in Tijms (2003a). It is the sum of a power series with monotonically reducing coefficients and exponential terms, and hence should have good convergence properties. From the rearranged form it can also be seen that the early coefficients also do not reflect the fact that the stage distribution has been truncated. It lends itself to efficient programming, so provided enough terms in the stage distribution have been calculated, the results should be fast and accurate. The constraint on the calculation appears to be Matlab's sensible refusal to calculate $e^{-\mu t}$ for values of $\mu t > 740$, so `mek1.m` will only calculate the waiting-time distribution out as far as the expected value of 740 stages of service.

Verification

The State Distribution. Programming the balance equations in Maple, which for rational values of λ and μ can calculate the probabilities as exact rational numbers, showed no error greater than 10^{-14} for a range of λ, μ, and k values, and out as far as 60 stages, when compared with `mek1.m`. As well as providing verification of the program, this showed that there is no evidence that the iterative calculation of the state probabilities is causing errors to build up excessively before the tail values of the distribution become very small.

The Waiting-Time Distribution. The Laplace-Stieltjes transform of the waiting-time distribution is

$$W_q^s = \frac{(1-\rho)s}{s-\lambda+\lambda(\mu/(\mu+s))^k}.$$

Maple is capable of inverting this (`invlaplace`) if given the Laplace-transform version (drop the "s" from the numerator), for moderate values of k, so, for example, for $k=2$ and $a=\lambda+4\mu$, $b=\sqrt{\lambda a}$,

$$P(w_q \le t) = 1-2\big((a(\lambda+\mu)\sinh(bt/2) + \lambda a\cosh(bt/2))e^{((\lambda-2\mu)t/2)}\big)/\mu a.$$

A comparison with the $M/M/1$ waiting-time distribution, and that for $k=2$, showed no difference greater than 10^{-12} out as far as the 99th percentile or $\mu=740$. For $k=5, 10, 20$, and traffic intensities up to 0.99 the maximum observed error was 10^{-6}, with much smaller error values for moderate traffic intensities. The results were also checked against QTSPlus and MCQueue (Tijms 2003b).

The Matlab Program. `mek1.m` is a Matlab.m file that calculates the state and waiting-time distributions using the method described above. All the input is from the screen and the main output is graphs of these distributions.

The Matlab diary function is used to also capture the coordinates for these graphs in files called State.txt and Waiting.txt, respectively, for exporting them to other packages.

The program can also be modified to a genuine finite capacity $M/E_k/1/N$ solver by making some minor changes to the section that calculates the stage distribution, reflecting the truncation of the balance diagram.

References

Bhat, U. N., G. K. Miller, S. S. Rao. 1997. Statistical analysis of queueing systems. J. H. Dshalolow, ed. *Frontiers in Queueing*, Chapter 13. CRC Press, Boca Raton, FL, 351–394.

Daley, D. J. 1968. The serial correlation coefficients of waiting times in a stationary single server queue. *J. Australian Math. Soc.* **8** 683–699.

Gross, D., C. M. Harris. 1974. 1985. *Fundamentals of Queueing Theory*, 1st and 2nd ed. John Wiley, New York.

Gross, D., J. F. Shortle, J. M. Thompson, C. M. Harris. 2008. *Fundamentals of Queueing Theory*, 4th ed. John Wiley, Hoboken, NJ.

Hall, R. W. 1991. *Queueing Methods for Services and Manufacturing*. Prentice Hall, Englewood Cliffs, NJ.

Ingolfsson, A., T. A. Grossman. 2002. Graphical spreadsheet simulation of queues. *INFORMS Trans. Ed.* **2**(2) 27–39. http://ite.pubs.informs.org/.

Law, A. M. 2007. *Simulation Modelling and Analysis*, 4th ed. McGraw-Hill, Boston.

Mandelbaum, A., S. Zeltyn. 2010. Service engineering: Data-based course development and teaching. *INFORMS Trans. Ed.* **11**(1) 3–19. http://ite.pubs.informs.org/.

McNickle, D. C. 1986. A practical queueing assignment. *N.Z. Oper. Res.* **14**(3) 101–104.

Tijms, H. C. 2003a. *A First Course in Stochastic Models*. Wiley, Chichester, UK.

Tijms, H. C. 2003b. MCQueue. http://staff.feweb.vu.nl/tijms/.

Wikipedia. 2011. http://en.wikipedia.org/wiki/Formative_assessment.

Welch, P. D. 1964. On a generalized $M/G/1$ queuing process in which the first customer of each busy period receives exceptional service. *Operations Res.* **12**(5) 736–752.