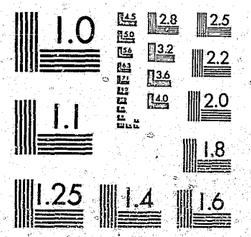


94715

94715 CI

DEVELOPMENT OF A DYNAMIC QUEUE-DEPENDENT DISPATCHING PROCEDURE

by

Chai-Yi Chou
under the supervision of
Professor James M. Tien

June 30, 1984

This report summarizes the doctoral thesis research effort that has been supported by the National Institute of Justice under Grant Number 82-IJ-CX-0037, awarded to Rensselaer Polytechnic Institute on October 1, 1982.

ABSTRACT

The motivation for this doctoral dissertation is a problem arising in the police dispatch area. Typically, citizens who call for police service -- for either a critical (i.e., requiring an immediate police response) or a non-critical (i.e., not requiring an immediate police response) matter -- are always being advised that "a patrol car will be right out", even though considerable delays may occur because of the unavailability of patrol cars, especially for responding to non-critical calls for service. Citizens are being needlessly frustrated; the frustration can be mitigated, if not eliminated, by formally advising citizens of potential delays. Indeed, because citizen satisfaction is a function of expectation and because some 86.1 percent of all calls for police service are non-critical in nature, a considerable portion of police demand can be "managed" and, more specifically, the formal delay procedure is one approach for managing such demand.

In 1976 the Wilmington Department of Police, Wilmington, Delaware, implemented a formal delay procedure; that is, when all patrol cars were busy, callers requesting service for a non-critical matter were told to expect a 30-minute delay. As an element of two consecutive patrol experiments, the formal delay procedure was evaluated and found to be very effective. It should, however, be noted that Wilmington's formal delay procedure is fixed or static; that is, callers receiving a formal delay are each advised of the same constant delay -- a 30-minute delay. Certainly, this need not and should not be the case. Depending on the state of the system, the expected delay should, of course, be variable and of different value for each non-critical caller. Thus, what is needed is a dynamic (i.e., state or queue dependent) procedure for delaying responses to non-critical calls. This then is the goal of the dissertation: the development of a dynamic delay procedure that could be straightforwardly implemented in any police department.

In analytical queuing terms, the dynamic delay procedure can be characterized as a prioritized queue-dependent model. The model is sensitive to the need to have enough patrol cars available to respond to critical calls, while at the same time not allow the non-critical calls to be queued up for too long. In addition to stating the problem that prompted the research and outlining a research approach consisting of eight explicitly defined activities, this summary report also provides a brief literature review and an exposition of some key results.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Motivation

The motivation for my dissertation is a problem arising in the police dispatch area. Tien and Valiante [1979] provide the following vivid description of the problem.

> A woman returns home at the end of an exhausting day at the office and finds her home in a disheveled state; it has been ransacked and burglarized. After taking stock of her losses and perhaps calling and commiserating with one or two of her close friends and relatives, she calls the police and is told, "A patrol car will be right out". Ten minutes pass, and no patrol car arrives. Ten more minutes pass, and still no patrol car. Because the late afternoon is a busy time and a platoon shift change could be occurring, a patrol car may not be available for dispatch to this *non-critical* (i.e., not requiring an immediate or emergency response) call-for-service for a rather long time, perhaps up to an hour from the time the call is received. Meanwhile, the woman is becoming increasingly distraught and frustrated -- her expectation, after all, was raised because she was told that a patrol car would be right out.

The above account is a common daily occurrence in cities throughout the nation. Citizens are always being advised that a "patrol car will be right out", even though considerable delays may occur either because no patrol cars are available for dispatch, or because the few cars that are available are being reserved for dispatch to more critical calls for service, or because the car that is assigned to the sector in which the call originated is busy. Whatever the reason, citizens are being needlessly frustrated. Certainly, the frustration can be mitigated, if not eliminated, by formally advising citizens of potential delays. Indeed, because citizen satisfaction is a function of expectation [Kansas City Police Department, 1977; Tien et al., 1978; Tien and Valiante, 1979] and because some 86.1 percent of all calls for police service are *non-critical* in nature [Tien et al., 1978; Sumrall et al., 1980], a considerable portion of police demand can be "managed" and, more specifically, the *formal delay procedure* is one approach for managing such demand.

In 1976 the Wilmington Department of Police (WDP), Wilmington, Delaware, implemented a formal delay procedure; that is, when all patrol cars were busy, callers requesting service for a non-critical matter were told to expect a 30-minute delay. As an element of both the Wilmington split-force patrol experiment [Tien et al., 1978] and the Wilmington management of demand program [Cahn and Tien, 1981], the formal delay procedure was judged to be very effective; the citizens' attitude toward a delay -- of which they were formally advised -- is best summarized by one of the telephone survey respondents who said, "I am a taxpayer. If it helps to keep my taxes down, then I'm all for the police to take their time in showing up to non-emergency situations -- but I would like to be told of such a delay so that I'm not just waiting around for them" [Tien and Valiante, 1979].

It should, however, be noted that Wilmington's formal delay procedure is fixed or static; that is, callers receiving a formal delay are each advised of the same constant delay -- 30-minute delay. Certainly, this need not and should not be the case. Depending on the state of the system (i.e., how many of the total number of patrol cars are busy, how many critical and non-critical calls are waiting in queue for service, at what rates the critical and non-critical calls for service are arriving at the police dispatch center, and how fast the patrol cars are handling the calls), the expected delay should, of course, be variable and of different value for each non-critical caller. Thus, what is needed is a *dynamic* (i.e., state or queue dependent) procedure for delaying responses to non-critical calls. This then is the goal of this dissertation: *the development of a dynamic delay procedure* Although such a procedure would be significantly enhanced by the availability of a computer-assisted dispatch (CAD) system, it is intended to develop a procedure that could be straightforwardly implemented in any police department.

1.2 Objective

In analytical queuing terms, the dynamic delay procedure can be characterized as a prioritized non-preemptive queue-dependent dispatching procedure. The system has a call-taker who receives calls for service and a dispatcher who dispatches the patrol cars on radio. The call-taker will be able to determine whether a call-for-service (CFS) is emergency or not, that is, a high priority call or a low priority call. When a CFS arrives, it will stay in queue until a patrol car arrives at the scene. This time is called the response time which is the sum of the delay time and the travel time. The delay time is the time elapsed since the CFS arrives until a patrol car is dispatched. The travel time is the time elapsed since a patrol car is dispatched until the car is on-scene. When the patrol car arrives it will spend some time on the scene, called the on-scene time. After that, it will be available for other CFS again. Exhibit 1.1 displays the time instants at which the CFS arrives, car is dispatched, car is on-scene and car is available again and defines the time intervals of the delay time, the travel time, the on-scene time, the response time and the service time.

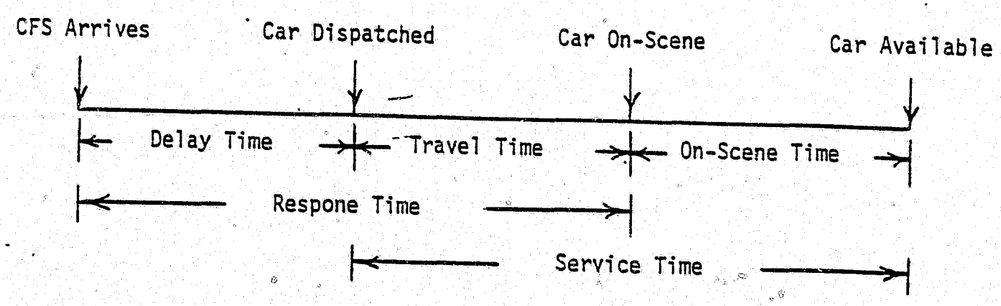


Exhibit 1.1

The objective of this study is to develop a queue-dependent dispatching procedure for determing the *conditional expected delay time of each call in queue given the state of the system and the position in queue* so that the probability of a high priority call has a delay is reduced and, at the same time, the delay of a low priority call is not too long. This can be achieved by reserving a few cars for high priority calls and restricting the length of the low priority queue when system is not full. We describe such a dispatching procedure in section 1.3.

1.3 Model definition

Suppose the total effective number of available patrol cars is N. If the call-taker receives a high priority (i.e., critical) call and at least one of the N total patrol cars is not busy, then the call-taker would inform the caller that a patrol car will respond with an expected delay time equal to the expected travel time. If, on the other hand, there is no patrol car available, then the high priority call is either lost or queued in the high priority queue. And if it is queued, the caller is advised of an expected delay equal to the expected waiting time in the high priority queue and the expected travel time. If the call-taker receives a low priority (i.e., non-critical) call and b, the number of busy patrol cars, is less than some number R, then the call-taker would inform the caller that a patrol car will respond with an expected delay time equal to the expected travel time. If, on the other hand, $b \geq R$ and the low priority queue length is less than some number M, then the low priority call is queued in the low priority queue and the caller is advised of an expected delay equal to the expected waiting time in the low priority queue and the expected travel time. In the case of $R \leq b < N$ and the low priority

queue length is equal to M, the call is queued at the end of low priority queue and a patrol car will be sent to the first call in the low priority queue. Whenever a patrol car becomes available, it will first attempt to serve the next high priority call if there is any; second, if $b \geq R$ and the low priority queue length is greater than M, it will attempt to serve the next low priority call, if there is any; otherwise, it will remain available until the next call arrives.

Notationally, the above dispatching procedure can be defined as: D(N;R,M) where

N = the total effective number of patrol cars.

R = cut-off point for the number of busy patrol cars; if the number of busy patrol cars is equal to or greater than R, then only high priority calls are served.

M = cut-off point for the number of calls in the low priority queue; if the number of calls in the low priority queue is equal to M when a low priority call arrives or when the number of calls in the low priority queue is greater than M when a patrol car becomes available, then the R cut-off does not apply and low priority calls will be served as long as there is no high priority calls awaiting to be served and as long as there is at least one patrol car available.

It can be seen that the D(N;R,M) procedure is quite mindful of the need to have enough patrol cars available to respond to high priority calls (i.e., the R cut-off), while at the same time not allow the low priority calls to be queued up for too long (i.e., the M cut-off).

In general, there are two models for the emergency calls. One is when it is required immediate response be made to the high priority call and, hence, the call will be lost if all N patrol cars are busy; the other is when the high priority call can be queued in the high priority queue and waits for its turn to be served. The low priority calls can always be queued. The D(N;R,M) procedure can apply to both of the two models. When D(N;R,M) applies to the first model (i.e., high priority calls are lost when all the patrol cars are busy) it is denoted as D(N;R,M;L,Q) which is discussed in Section 2; the L indicates that high priority calls are lost when the system is totally busy and Q indicates that the low priority calls can be queued. When D(N;R,M) applies to the second model it is denoted as D(N;R,M;Q,Q) which is discussed in Section 3; the two Q's indicate that both the high and low priority calls can be queued.

1.4 Literature Review

There is, of course, an immense literature dealing with queueing. But, suprisingly, very few articles have considered the D(N;R,M) procedure and all of those articles which considered the D(N;R,M) procedure are special cases of the model we propose here. Those articles related to the D(N;R,M) procedure are summarized in the Exhibit 1.2 and discussed below. In addition to those articles, we also looked into two approximate solution methods, the fluid approximation and the diffusion approximation and discuss the applicability of these approximation methods to our proposed queuing system.

The D(N;N,∞;L,Q) model is essentially equivalent to the M/M/N queue and is widely referenced in the literature.

The cutoff priority queuing model was first introduced by Benn [1966] in his dissertation and summarized in Jaiswal's Priority Queue [1968]. For the

Key Variable / Model	Distribution of Delay Time of Low Priority Calls	Expected Delay Time of Low Priority Calls	Distribution of Server Utilization	Distribution of System States	Conditional Expected Delay Times
D(N;N,∞;L,Q)	*	*	*	*	*
D(N;R,∞;L,Q)	Taylor & Templeton [1980]	Taylor & Templeton [1980]	Jaiswal [1968]	Sonick & Jackson [1973]	Proposed To Do
D(N;R,M;L,Q)	--	Proposed To Do Taylor & Templeton [1976] -- only for R=N-1	Proposed To Do Taylor & Templeton [1976] -- only for R=N-1	Proposed To Do	Proposed To Do
D(N;N,∞;Q,Q)	Davis [1966]	Cobham [1954]	Davis [1966]	Proposed To Do	Proposed To Do Dressin & Reich [1957] -- for N=1 only
D(N;R,∞;Q,Q)	Taylor & Templeton [1980]	Taylor & Templeton [1980]	Jaiswal [1968]	Proposed To Do	Proposed To Do
D(N;R,M;Q,Q)	--	Proposed To Do	Proposed To Do	Proposed To Do	Proposed To Do

*The D(N;N,∞;L,Q) model is essentially the M/M/N queue, which has been extensively dealt with in the literature.

Exhibit 1.2 Summary of Literature Review

D(N;R,∞;L,Q) model, Jaiswal [1968] used the technique of grouping the states which have the same number of busy servers to obtain the server utilization probability, the probability of low priority call being delayed and the probability of high priority call lost at equilibrium. Sonick and Jackson [1973] used an iterative method to generate empirically the distribution of queue length (up to some finite number) at equilibrium. Taylor and Templeton [1980] solved the distribution of unconditional delay time of low priority calls analytically by the transform technique and, then, computed the expected delay time.

For the D(N;R,M;L,Q) model, Taylor and Templetion [1976] worked on a special case in which R=N-1. For this special case, they used the transform technique to obtain the server utilization probability, the expected queue length of low priority call and, then, using Little's formula, to compute the unconditional expected delay time of low priority call. No work has been done for the general model.

For the D(N;N,∞;Q,Q) model, Cobham [1954] used busy period analysis on the high priority call to obtain the unconditional expected delay time for the low priority call. Dressin and Reich [1957] obtained some results for single server queuing system which allows customers of more than two levels of priority; they formed the balance equations and analyticall solved the probability distribution of the number of customers of priority p or higher in queue at equilibrium. And, also, they obtained the conditional density function for the delay of a customer of priority p given that a priority p customer arrives and finds the server is busy and has n customers of priority p or higher waiting in the queue. Then, they weighted the conditional delay time density function by the corresponding steady state probabilities to get the unconditional delay time density function of priority p

customer (in the transform space). Davis [1966] extended Dressin and Reich's work to a multi-server queuing system. By doing the same analysis as Dressin and Reich, he derived the probability that all the N servers are busy and there are k calls of priority p or higher in queue when a call of priority p arrives. Further, he used the same conditional delay time density function as in Dressin and Reich's to get the unconditional delay time density function of priority p call.

For the D(N;R,∞;Q,Q) model, Jaiswal [1966] used the discrete transform technique to obtain the server utilization probability and to compute the probabilities of high priority and low priority calls being delayed, respectively. Taylor and Templeton [1980] used a different method, matrix iteration, to obtain the server utilization probability and the distribution of unconditional delay time of low priority calls.

For the D(N;R,M;Q,Q) model, no literature exists at present.

Two potentially pertinent solution methods are briefly discussed next.

Newell [1965] proposed an approximation method, the fluid approximation, to solve some practical queuing problems which have large queues and long delays. The fluid approximation assumes that the cumulative number of arrivals, $\alpha(t)$, can be approximated by a *non-random continuum* as if it were a fluid flowing into a reservoir and the cumulative number of departure, $\delta(t)$, can be approximated by a *non-random continuum* as if it were a fluid flowing out of a reservoir. That is, it disregards the stochastic effects and uses the mean values as the estimates. Also, in order to maintain the independence of the input and output flow, one constraint has to be satisfied, i.e., the queue length should not drop to zero during the time period of analysis. N(t), the blacklog in the system expressed in terms of the number of customers

at time t, is equal to $\alpha(t) - \delta(t)$. So, N(t) is also a continuous process. The fluid approximation is not suitable to analyse the proposed problem for three reasons. First, the queue of the proposed problem is formed not only because of the high arrival rate but also because of the fluctuation of randomness. Second, the utilization factor of the proposed problem is never greater than 1. Hence, if the fluid approximation method were applied, it will give zero queue length during the times at which the non-zero queue constraint is violated. Third, due to the continuum of N(t), the fluid approximation is not able to find the conditional expected delay time of a call given its position in queue and this is the biggest drawback of using the fluid approximation to approach our proposed problem.

Gaver [1968] and Newell [1968] used a better approximation method, the diffusion approximation, to solve problems in some heavily loaded system. In the diffusion approximation, it is assumed that the arrival process and the departure process are both approximated by *continuous* random process which at time t are normally distributed with mean $\alpha(t)$ and $\delta(t)$ and variance $\sigma^2_{\alpha(t)}$ and $\sigma^2_{\delta(t)}$, respectively. The variance terms are introduced in order to represent the random fluctuation of these processes about their mean. N(t), the backlog in the system expressed in terms of the number of customers at time t, is equal to $\alpha(t) - \delta(t)$ and it is a continuous process. The diffusion approximation is also not suitable to analyse the proposed problem for two reasons. First, the diffusion approximation gives a reasonable good estimate only when the system utilization is greater than .9 [Gaver, 1968]. For our proposed problem, a significant number of low priority calls may be waiting in queue for service but this is due to the D(N;R,M) queue discipline and is not due to the heavy load. Second, because of the continuum of N(t), the

diffusion approximation is not able to find the conditional expected delay time given its position in queue.

1.5 Scope of Work

The primary objective of this dissertation is to develop a numerical solution algorithm of the contional expected delay times for low priority calls for each of the D(N;R,M;L,Q) and the D(N;R,M;Q,Q) models and to validate our numerical algorithms for the special cases by analytical means. A GPSS (General Purpose Simulation System) simulation is also undertaken so that we can check our numerical algorithms more generally.

The eight explicityly defined activities that constitute this thesis research are outlined below -- they are stated in proposal terms at this time. Then, in Sections 2 and 3, we provide the key results of the research. In Section 2, we develop the D(N;R,M;L,Q) model and 3 cases were considered: 1) R=N and M=∞, 2) R<N and M=∞, and 3) R<N and M is finite. For each of these three cases, we develop two algorithms, the steady state probability distribution algorithm and the conditional expected delay time for non-emergency calls algorithm, using the Markovian assumptions. The unconditional expected delay time are also computed. Case 1 is trival because it is equivalent to the M/M/N queuing system. Case 2 only has one cut-off point (i.e., R), the cut-off on the number of busy servers, and there is no restriction on the length of low priority queue. In other words, it is a strict cut-off dispatching procedure. The conditional expected delay time of Case 2 can be used as an upper bound to that of Case 3, the general model. The unconditional expected delay times are checked with the analtyical results. In Case 3, there are two cut-off points (i.e., R and M). The two algorithms in Case 3 have been coded in FORTRAN. For R=N-1, the unconditional expected delay times are checked with the analytical results. Several properties are also observed and explained.

In Section 3, the D(N;R,M;Q,Q) model is developed. As in Section 2, we consider the same three cases. We also develop two algorithms for each case and, then, the unconditional expected delay time is computed. Two theorems and several lemmas are proved for Case 3, the most general model. In Case 2, the unconditional expected delay time agrees with the analtyical result of Taylor and Templeton's analytical solution. In Case 3, we check the limiting values (i.e., when $M\to\infty$) of the unconditional expected delay times; they also agree with the analytical solution. Finally, we make comparisons between the two models, D(N;R,M;L,Q) and D(N;R,M;Q,Q), and the results seem reasonable.

In Sections 2 and 3, the reader is cautioned about the fact that we use the conventional queuing terms; that is, we use waiting time instead of delay time, customer instead of CFS, server instead of patrol car, etc. The thesis itself employs application-oriented rather than queuing-oriented terms.

Activities 1 and 2: Develop the D(N;R,M;L,Q) and the D(N;R,M;Q,Q) Models and the Corresponding Numerical Solution Algorithms

As we can see from Exhibit 1.2, almost all of the articles on the D(N;R,M) procedure are special cases of our proposed models, the D(N;R,M;L,Q) and the D(N;R,M;Q,Q) models. And almost all of those articles emphasize system wide measures such as the distribution of server utilization, the unconditional expected delay time of each type of calls. These are useful in planning the system but of little use in operating the system. Except for Dressin and Reich's paper [1957], no other paper dealt with the *conditional expected delay time* of each waiting call in queue given the current state of the system and the position in queue. The conditional expected delay time is very important because under the D(N;R,M) dispatching procedure a significant number of calls will have a delay, especially the low priority calls which are the majority of the calls for service. So, it would be important to tell the caller how

long he/she is expected to wait until a patrol car arrives. And also, for those callers who called earlier and are still waiting for the patrol car to show up, we can give them the updated expected response times if and when they call again.

So, we would like to develop a solution algorithm for each of the D(N;R,M;L,Q) and the D(N;R,M;Q,Q) models which will give us the conditional expected delay time of each call in queue given the system state and the position in queue. Initially, due to the complexity of the model, we need two Markovian assumptions to simplify the development of the algorithms. The two Markovian assumptions we made are i) the arrival of the high and the low priority calls are independent homogeneous Poisson processes with rates λ_1 and λ_2, respectively, ii) the service time (for either types of call) is exponentially distributed with a constant expected service time of $1/\mu$. In addition to the two Markovian assumptions, we also made another assumption, that is, iii) each call receives the services of only one car. Define ρ_1, ρ_2 and ρ as follows: $\rho_1=\lambda_1/\mu$, $\rho_2=\lambda_2/\mu$ and $\rho=\rho_1+\rho_2$.

Since the arrival rates λ_1 and λ_2 and the service rate μ are constant in time the conditional expected delay times are time-invariant under the assumptions made. That is, the values generated by the algorithm are valid for all times during system operation (both the transient and the steady state periods).

The *unconditional expected delay time* can be computed by summing the product of the conditional expected delay times and the corresponding steady state probabilities of each state. The reason we want to compute the unconditional expected delay time is that it will help us to validate our algorithm by analytical means. Hence we need to develop another solution algorithm which will generate the steady state probabilities numerically.

Activity 3: Validate Special Cases by Analytical Means

For the D(N;R,∞;L,Q) model, i.e., the D(N;R,M;L,Q) model with M=∞, Taylor and Templeton [1980] had the analytical solution of the unconditional expected delay time for the low priority call. For the D(N;N-1;M;L,Q) model, i.e., the D(N;R,M;L,Q) model with R=N-1, Taylor and Templeton [1976] also had the analytical solution of the unconditional expected delay time for the low priority call. We would like to check our numerical algorithm for the same special conditions.

For the D(N;R,∞;Q,Q,) model, i.e., the D(N;R,M;Q,Q) model with M=∞, Taylor and Templeton [1980] obtained the analytical expression of the unconditional expected delay time for the low priority call. Our value for the case of M=∞ should check with theirs. For the D(N;R,M;Q,Q) model with M equal to a finite number, there is no literature available to date. We would attempt to solve the special case of R=N-1 analytically. Hopefully, we would be successful so that we can use it to check our numerical algorithm. Note that the D(N;R,∞;Q,Q) model is the limiting model of D(N;R,M;Q,Q) model when M→∞. Since the analytical solution is available for the D(N;R,∞;Q,Q) model, we can check our numerical algorithm for the D(N;R,M;Q,Q) model when M is large.

Activity 4: Validate More Generally by Simulation

As we can see in activity 3, we only can validate the special case of our algorithms through the analytical means. In order to validate them more generally, a GPSS simulation is proposed. In this simulation, we will choose the CFS as the transaction and the patrol car as the server.

We want to measure the conditional expected delay time under the rules of D(N;R,M) dispatching procedure and the three assumptions we made earlier. To measure the conditional expected delay time we have to take down following information at the instant transaction arrives: 1) type of the transaction, 2) the number of busy servers, 3) the number of transactions in the high

priority queue and 4) the number of transactions in the low priority queue. We can foresee that, for reasonable values of R and M and moderate input and output rates, a significant number of transactions will have zero delay and the probability of a large number of transactions in the system at the same time will be very small if we start it out with an empty system. This means that we need to run the simulation program for a long period of time to get enough data to validate our algorithm. Fortunately, there is an easy way to get around this. Since the conditional expected delay time is time-invariant, we can start it out with an extremely busy system. Thus, it will not take too long to get enough information to validate our conditional expected delay times algorithm.

Activity 5: Determining the Impact of Relaxing Model Assumptions by Numerical and Simulation Methods

Referring to the activities 1 and 2, we remind ourselves that the numerical algorithms are developed under the three assumptions made there. They are 1) the arrival processes are independent homogeneous Poisson processes, 2) the service time (sum of the travel time and the on-scence time) is exponentially distributed with a constant rate and 3) each call receives service by only one car. We would like to relax these assumptions to make the model more realistic and applicable. Taylor [1976] and others have addressed the first two assumptions. The independence of the two arrival processes seems to be valid since there is no relation between the emergency and non-emergency calls. Further, the emergency calls can be described by a Poisson process and if we narrow the period of analysis, for example 4 periods per day, the rate can be considered as a constant. Because of the lack of data on the non-emergency calls, he did not test the arrival pattern of non-emergency calls. But there is good reason to assume that it is also a homogeneous Poisson process. Because

a large proportion of the non-emergency calls are not scheduled and since they arrive independently from different sources their combined arrival pattern tends to be a Poisson process. The reason for the homogeneity of non-emergency calls is the same as in the case of the emergency calls. The assumption of exponential service time is not appropriate here. The best-fit distribution is the Erlang distribution of order k, k>1.

Green [1978, 1980, 1981] has addressed the third assumption. She considered a class of queues which is characterized by customers who require simultaneous service from a random number of servers. One of her models is that a customer cannot begin service until all required servers are available and once the service begins each individual server will have independent service completion time, that is, the servers do not necessarily end service together. We would like to adopt this assumption together with the Erlang-k (with k=2,3 and ∞) service completion time assumption to develop numerical solution algorithms for the steady state probability and the conditional expected delay time. Also, under the new assumptions, we would like to write a GPSS simulation program to check our numerical solution algorithms.

Activity 6: Perform Sensitivity Analysis Using Numerical Approach

Up to now, we are assuming the arrival rates are constant during a time period. But it is not so in the real world. It will have some (maybe small) fluctuations in time. We would like to perform the sensitivity analysis on the arrival rates using a numerical approach. In order to do this we have to specify two measures about the service level. The first measure we choose is the probability that a high priority call has a zero delay at equilibrium. The second measure is the conditional delay time of the M^{th} low priority call in queue when all the servers are busy, i.e., the system is full. The reason

for choosing the first measure is obvious. And, the reason to choose the second measure is that it is an upper bound for the maximum expected delay that a low priority call can experience under the D(N;R,M) dispatching procedure when there is a car available. We would like to examine the sensitivity of these two measures with respect to changes in the total arrival rates for fixed N, R and M. Both of these measures will be obtained from the solution algorithms in Activity 5.

Activity 7: Develop Approximate Numerical Algorithms and Validate by Analytical and Simulation Methods

The numerical solution algorithm we propose to develop in Activity 5 is based on an analytical formulation. That is, it will produce the exact values of the conditional expected delay times for each low priority call in queue. What we really want in practice is an algorithm which will produce an approximate and acceptable answer very fast. The "acceptable answer" should be within the range specified by a lower bound and an upper bound for a given number of busy servers. We would like to investigate the bound analytically and by simulation. For fixed N, R and M, we plot the bounds of the expected delay times against the number of busy servers. If the bounds are tight, we can simply take the average as an approximate value for all the M calls. If the bounds are not tight, we will try to interpolate these M values by a curve between the bounds.

Activity 8: Refine Numerical Algorithms for use in Both Manual and Automated Environment

Because we hope to see that these algorithms are implemented by the police departments or other emergency service systems, we will refine the algorithms so that it can be used in both a manual and an automated environment. For

those systems which have computer (perhaps, microcomputer) facilities they can implement our computer-based algorithms directly on their facility. Then, by entering the model parameters N, R, M, the current rates λ_1, λ_2, μ and the current state of the system, it will determine the expected delay times for each of the non-emergency call in queue. For those systems which are not in an automated environment, we would like to produce some easy-to-read plots of the conditional expected delay times so that when a CFS arrives the call-taker can give the caller an estimated response time.

2. D(N;R,M;L,Q) Model

In the D(N;R,M;L,Q) dispatching procedure, there is only one queue, the low priority queue. The high priority customer will leave the system without having service when all the servers are busy at the time when he/she arrives. The state of the system can be represented as (n,q), where n is the number of busy servers and q is the number of low priority customers waiting in the queue. In this section we assume that both of the high priority customer and the low priority customer have Poisson arrival time with the arrival rates λ_1 and λ_2 respectively and the servers have exponential service time with the same expected service time $1/\mu$ for both types of customers, $\rho_1=\lambda_1/\mu$, $\rho_2=\lambda_2/\mu$ and $\rho=\rho_1+\rho_2$.

There are three cases we should consider. The first two are the special cases of the third one.

(1) R=N and M=∞

(2) R<N and M=∞

(3) R<N and M is finite

We will study for each case the steady state probability distribution and the conditional expected waiting times of low priority customers at a given state.

Let P(n,q) denote the steady state probability at state (n,q), $EW_M^h(n,q;k)$ denote the conditional expected waiting time of the k^{th} high priority customer at state (n,q) in D(N;R,M;L,Q) model and $EW_M^\ell(n,q;k)$ denote the conditional expected waiting time of the k^{th} low priority customer at state (n,q), $k\leq q$, in D(N;R,M;L,Q) model. The unconditional expected waiting time is denoted by EW_M^h and EW_M^ℓ for high priority customer and low priority customer respectively.

2.1 D(N;N,∞;L,Q) Model

For this dispatching procedure, no server is reserved for high priority customer. When the system is not full, we send a server to the just arrived customer no matter what the priority the customer is. When the system is full, the high priority arrivals are lost and the low priority arrivals are queued in the low priority queue and wait for the service. When a server completes a service and returns to free, it will check the low priority queue. If the queue is not empty then the first waiting low priority customer will start the service. If the queue is empty, the server will remain free. Exhibit 2.1 is the transition diagram for D(N;N,∞;L,Q) model.

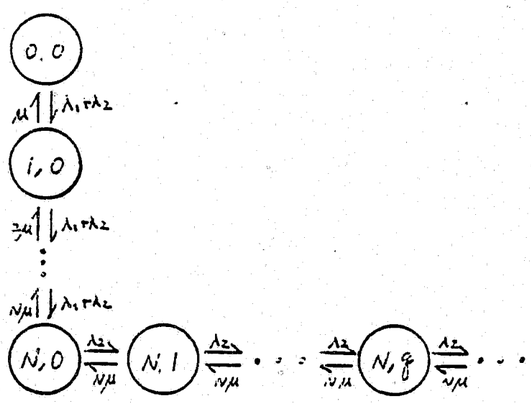


Exhibit 2.1 Transition diagram for D(N;N,∞;L,Q) model

Steady State Probability Distribution

From the transition diagram in Exhibit 2.1, we can recognize that it is M/M/N queue with input rate $\lambda_1+\lambda_2$ and service rate μ per server before the system is full. When the system is full, it changes to M/M/1 queue with input rate λ_2 and service rate $N\mu$. Hence, it is very easy to write down the steady state probability P(n,q).

$$P(n,0) = \frac{\rho^n}{n!} P(0,0), \quad n=1,2,\ldots N \tag{2.1}$$

$$P(N,q) = \left(\frac{\rho_2}{N}\right)^q P(N,0)$$

$$= \left(\frac{\rho_2}{N}\right)^q \frac{\rho^N}{N!} P(0,0), \quad q=1,2,\ldots \tag{2.2}$$

Summing all the probabilities to 1, we have

$$P(0,0) = \left[\sum_{n=0}^{N-1} \frac{\rho^n}{n!} + \frac{N}{N-\rho_2} \frac{\rho^N}{N!}\right]^{-1} \tag{2.3}$$

Substitute (2.3) to (2.1) and (2.2), we will have all the steady state probability. Note that the steady state can be reached only when $\rho_2<N$.

The Conditional Expected Waiting Times for High Priority Customer

Since the high priority is not allowed to be queued in D(N;N,∞;L,Q) model, the conditional expected waiting time of high priority customer will be 0 at any state.

The Conditional Expected Waiting Times for Low Priority Customer

Let W_k^{ℓ} be the time that the k^{th} low priority customer spent in the queue, k=1,2,... By the strategy of this dispatching procedure, only when the system is full the low priority customer starts to queue up and at this time the high priority customer is not allowed to enter the system. Hence, the k^{th} low priority customer will start the service until a server completes the service k times. W_k^{ℓ} is the Erlang distribution with parameters $N\mu$ and k. The density function is

$$f_{W_k^{\ell}}(t) = \frac{(N\mu)^k}{(k-1)!} t^{k-1} e^{-N\mu t}, \quad k=1,2,\ldots \tag{2.4}$$

The expected waiting time of the k^{th} low priority customer at state (N,q) is

$$EW_{\infty}^{\ell}(N,q;k) = \frac{k}{N\mu}, \quad q \geq k \geq 1 \tag{2.5}$$

The unconditional expected waiting time for low priority customer is

$$EW_{\infty}^{\ell} = \sum_{k=0}^{\infty} P(N,k)\, EW_{\infty}^{\ell}(N,k+1;k+1) \tag{2.6}$$

The reason for equation (2.6) is, when a low priority customer arrives and finds the system at state (N,k) then he/she will enter the low priority queue at the $(k+1)^{st}$ position. So, the expected waiting time for him/her is $EW_{\infty}^{\ell}(N,k+1;k+1)$.

2.2 D(N;R,∞;L,Q) Model with R<N

For this dispatching procedure, N-R servers are reserved for high priority customer. That is, when the number of busy servers, b, is less than R the arriving customer will be served immediately no matter what his/her priority is. But when $b \geq R$, only high priority arrivals are allowed to be served immediately and low priority arrivals have to join the low priority queue. When b=N, i.e., the system is full, the high priority arrivals will be lost. A low priority in the queue can be served only when b drops below R and the queue discipline is FIFO. Exhibit 2.2 is the transition diagram for D(N;R,∞;L,Q) model.

Steady State Probability Distribution

In Jaiswal's priority queues [1968], it has a formula to compute P(0,0) for this model.

$$P(0,0) = \left[\sum_{n=0}^{R-1} \frac{\rho^n}{n!} - \frac{\rho^R}{(R-1)!\rho_2} \left\{ \frac{(\rho_2 - R)}{\rho_2} \frac{\rho_1^R}{R!} + \sum_{n=R+1}^{N} \frac{\rho_1^n}{n!} \right\}^{-1} \sum_{n=R}^{N} \frac{\rho_1^n}{n!} \right]^{-1} \tag{2.7}$$

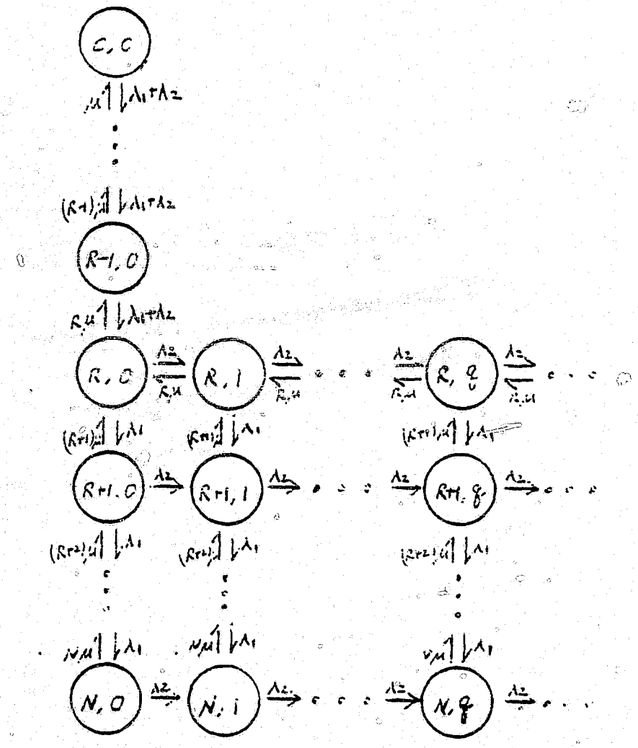


Exhibit 2.2 Transition diagram for D(N;R,∞;L,Q) model

We will use (2.7) to compute the steady state probabilities. We summarize the steps below:*

STEP 1) Compute P(n,0), n=1,2,...R.

STEP 2) Define a recursive sequence to compute P(n,0), n=R+1,...N.

STEP 3) For each $q \geq 1$, do

STEP 3.1) Compute P(R,q)

STEP 3.2) Define two recursive sequences to compute P(n,q), n=R+1,...N.

*In this proposal, we will forego presenting the detailed derivation of our results.

We outline these steps below.

STEP 1) Compute P(n,0), n=1,2,...R.

Before the number of busy servers reaches R, it is M/M/R queue. Hence,

$$P(n,0) = \frac{\rho^n}{n!} P(0,0) , \quad n=1,2,...R \tag{2.8}$$

STEP 2) Define a recursive sequence to compute P(n,0), n=R+1,...N.

Form the balance equation at state (N,0),(N-1,0),...(R+1,0) successively, we obtain a recursive sequence A_n, n=N,N-1,...R,

$$A_n = [(n+1+\rho)A_{n+1} - (n+2)\ A_{n+2}]/\rho_1 \tag{2.9}$$

with A_N=1 and $A_{N-1} = (N+\rho_2)/\rho_1$.

Then the steady state probability at state (n,0), n=R+1,...,N is

$$P(n,0) = (A_n/A_R)\ P(R,0), \quad n=R+1,...N \tag{2.10}$$

STEP 3) For q≥1 do

Step 3.1) Compute P(R,q)

$$P(R,q) = (\rho_2/R) \sum_{n=R}^{N} [P(n,q-1)] \tag{2.11}$$

Step 3.2) Define two recursive sequence to compute P(n,q), n=R+1,...N.

Form the balance equation at state (N,q),(N-1,q),...(R+1,q) successively. We obtain two recursive sequence A_n, B_n, n=N,N-1,...R,

$$A_n = [(n+1+\rho)\ A_{n+1} - (n+2)\ A_{n+2}]\rho_1$$

$$B_n = [(n+1+\rho)\ B_{n+1} - (n+2)B_{n+2} - \rho_2 P(n+1,q-1)]/\rho_1 \tag{2.12}$$

with $A_N = 1$

$A_{N-1} = (N+\rho_2)/\rho_1$

$B_N = 0$

$B_{N-1} = -\rho_2 P(N,q-1)/\rho_1$

Then

$$P(n,q) = (A_n/A_R)[P(R,q) - B_R] + B_n, \quad n=R+1,...,N \tag{2.13}$$

We can increase q by 1 and repeat STEP 3.

We do a simple example to illustrate this procedure.

Example 2.1: Find the steady state probability distribution in D(5;3,∞;L,Q) model with λ_1=1, λ_2=1, μ=1.

We have N=5, R=3, $\rho_1=\rho_2$=1, ρ=2

Plug these values in (2.7), we have

P(0,0) = .12409

At STEP 1, by (2.8), we have

P(1,0) = .24818

P(2,0) = .24818

P(3,0) = .16545

At STEP 2, by (2.9), we have

$A_5 = 1$, $A_4 = 6$, $A_3 = 31$

Then, by (2.10), we have

P(4,0) = .03202

P(5,0) = .00534

For q=1, by (2.11) at STEP 3.1), we have

P(3,1) = .0676

At STEP 3.2), by (2.12), we have

$A_5 = 1$, $B_5 = 0$

$A_4 = 6$, $B_4 = -.00534$

$A_3 = 31$, $B_3 = -.06406$

By (2.13), we have

$P(4,1) = .02014$

$P(5,1) = .00424$

We can keep increasing the q until we find enough state we want.

In writing computer program one may choose an ε so that the total probability found so far is greater than $1-\varepsilon$ to stop the iteration on q.

Note that the sequence A in the algorithm will not change at each iteration and the steady state can be reached only when $\rho_2 < R$.

The Conditional Expected Waiting Times for High Priority Customer

The conditional expected waiting time for high priority customer is 0 at any state because no high priority is allowed to be queued in this dispatching procedure.

The Conditional Expected Waiting Times for Low Priority Customer

In this model the conditional waiting time distribution for low priority customer is not known. However, we found a very easy way to compute the expected value. Let $EW_\infty(n,q;k)$ denote the conditional expected waiting time of the k^{th} low priority customer at state (n,q) in D(N;R,∞;L,Q) model, where $k \le q$. Here we have dropped the superscript 2 without confusion. The expected waiting time for a low priority only depends on the position in the queue and does not depend on how many low priority in the queue. Notationally, this property can be written down in the following equation

$$EW_\infty(n,q;k) = EW_\infty(n,k;k), \quad q \ge k \ge 1 \tag{2.14}$$

This property is true only when M=∞, i.e., no restriction on the low priority queue. When M is finite, equation (2.14) will not hold any more. Because at the time the queue length reaches M, the system will have one more server to work and the service rate of the system will be bigger than it was. We will discuss this more in Section 2.3.

We now derive a recursive formula to compute $EW_\infty(n,q;k)$. The system will stay in the current state until one of the following three events occurs:

(A) A server completes service and returns to free.

(B) A high priority customer arrives.

(C) A low priority customer arrives.

As soon as one of the three events occurs the system will change the state. Now, let's compute $EW_\infty(n,q;k)$ for k=1. At state (N,1),

$$EW_\infty(N,1;1) = \frac{1}{N\mu+\lambda 2} + \frac{N\mu}{N\mu+\lambda 2} EW_\infty(N-1,1;1) + \frac{\lambda 2}{N\mu+\lambda 2} EW_\infty(N,2;1) \tag{2.15}$$

The first term on the righthand side (RHS) of (2.15) is the expected duration time at state (N,1). The 2nd term on the RHS of (2.15) is the product of the probability of event (A) occurs and the expected waiting time of the 1st low priority customer when even (A) occurs. The third term on the RHS of (2.15) is the product of the probability that event (C) occurs and the expected waiting time of the 1st low priority customer when even (C) occurs. Note that event (B) cannot occur at state (N,1). Apply (2.14) and multiply both sides by $N\mu+\lambda 2$, we have

$$EW_\infty(N,1;1) = EW_\infty(N-1,1;1) + \frac{1}{N\mu} \tag{2.16}$$

Continue to do this at state (N-1,1),(N-2,1),...and (R,1) successively, we will have

$$EW_\infty(R,1;1) = D_R$$

$$EW_\infty(n,1;1) = EW_\infty(n-1,1;1) + D_n, \quad n=R+1,\ldots,N \tag{2.17}$$

where D_n is defined recursively in the following way

$$D_n = \frac{1+\lambda_1 D_{n+1}}{n\mu}, \quad n=N-1,N-2,\ldots,R$$

$$\text{with } D_N = \frac{1}{N\mu} \tag{2.18}$$

By the same analysis, we can extend this to $k \geq 1$.

$$EW_\infty(R,k;k) = kD_R$$

$$EW_\infty(n,k;k) = EW_\infty(n-1,k;k) + D_n, \quad n=R+1,\ldots,N \tag{2.19}$$

where the D_n's are defined recursively as in (2.18).

Note that equation (2.19) together with (2.14) has defined all the expected waiting times of low priority customer at all the feasible states. We do the example 2.1 to illustrate how this recursive relation works. In example 2.1, we have N=5, R=3, λ_1=1, λ_2=1 and μ=1.

By (2.18), we have

$D_5 = .2$

$D_4 = .3$

$D_3 = .433$

By (2.19), we have, for $k \geq 1$

$EW_\infty(3,k;k) = .433\ k$

$EW_\infty(4,k;k) = .433k + .3$

$EW_\infty(5,k;k) = .433k + .3 + .2 = .433k + .5$ Q.E.D.

Taylor and Templeton [1980] found the unconditional expected waiting time of low priority customer for this model. We would like to compare our result to theirs. In order to make the comparison, we have to unconditionalize our conditional expected waiting times. To do this we have to weigh our conditional expected waiting times by the corresponding steady state probabilities and sum them all up. The unconditionalizing formula is

$$EW_\infty = \sum_{j=0}^{\infty} \sum_{i=R}^{N} P(i,j)\ EW_\infty(i,j+1;j+1) \tag{2.20}$$

The reason for (2.20) is when a low priority arrives and finds the system at state (i,j) then he/she will join the low priority queue (if necessary) in the $(j+1)^{st}$ position. And, the expected waiting time for him/her will be $EW_\infty(i,j+1;j+1)$.

A program has been written which generates P(i,j), $EW_\infty(i,j;j)$ and computes EW_∞ in (2.20), with j up to 49. Exhibit 2.3 is a list of the comparisons of the expected waiting time for low priority customer between the theoretical value and the result obtained from the program.

As we can see from Exhibit 2.3, our result checks with the theoretical value. As expected, our result is not greater than the theoretical value because in the program it only includes part of equation (2.20). The conditional expected waiting time of D(N,R,∞;L,Q) can be used as upper bound for D(N;R,M;L,Q) model. We will discuss this in Section 2.3.

2.3 D(N;R,M;L,Q) Model with R<N and M<∞

For this dispatching procedure, we reserve N-R servers for high priority customer and, at the same time, we do not allow the low priority queue to exceed M when there is a server free. The state space, S, is

N	R	λ_1	λ_2	μ	Expected Waiting Time	
					Theoretical Value	Our Result
5	3	1	1	1	.3418	.3418
5	4	1	2	1	.3649	.3649
5	4	2	1	1	.2246	.2246
10	8	1	1	1	.0002	.0002
10	8	2	4	1	.1679	.1678
10	8	4	2	1	.1263	.1263
15	12	1	1	1	0	0
15	12	2	8	1	.2272	.2271
15	12	8	2	1	.142	.142
15	12	.05	.25	.034	2.3067	2.3067

Exhibit 2.3 Comparison of the unconditional expected waiting times in D(N;R,∞;L,Q) model

$$S = \left\{ \begin{array}{lll} (n,q): & q=0 & \text{when } n=0,1,\ldots,R-1 \\ & q=0,1,\ldots M & \text{when } n=R,\ldots,N-1 \\ & q=0,1,\ldots & \text{when } n=N \end{array} \right\}$$

When a high priority customer arrives and finds the system at state (n,q), he/she will be served immediately if n<N, otherwise he/she will leave the system without having the service. When a low priority customer arrives and finds the system at state (n,q), then exactly one of the following four actions will be taken:

1) if n<R, he/she will be served immediately.
2) if R≤n<N and q<M, he/she will join the queue at the last position.
3) if R≤n<N and q=M, he/she will join the queue at the last position and the first waiting low priority customer in the queue will start the service immediately.

4) if n=N, he/she will join the queue at the last position.

When a server completes a service and the system changes state from (n,q) to (n-1,q) then exactly one of the following four actions will be taken:

1) if n-1<R and q=0, it will remain free.
2) if n-1<R and q>0, the first waiting low priority customer will start the service and the system will change the state from (n-1,q) to (n,q-1) instantaneously.
3) if n-1≥R and q≤M, it will remain free.
4) if n-1≥R and q>M, the first waiting low priority customer will start the service and the system will change the state from (n-1,q) to (n,q-1) instantaneously.

Exhibit 2.4 is the transition diagram for D(N;R,M;L,Q) model.

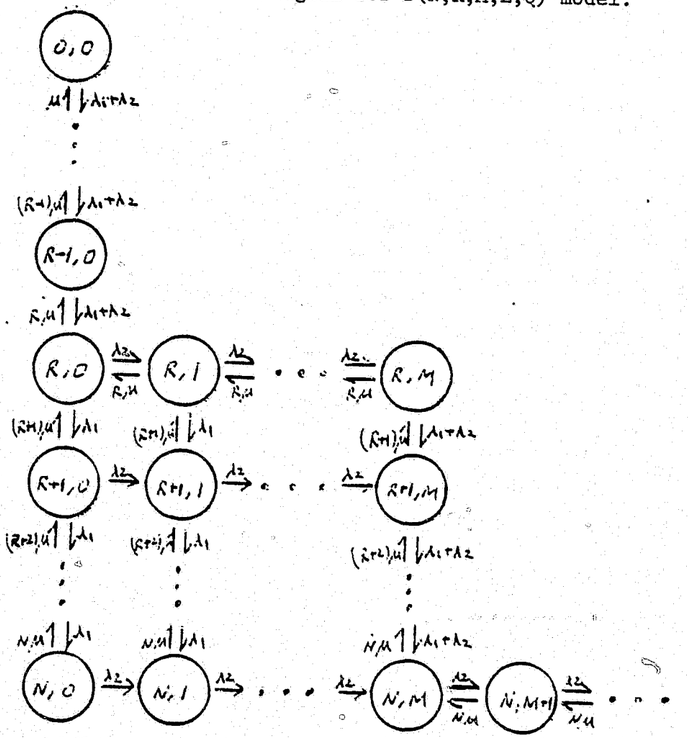


Exhibit 2.4 Transition diagram for D(N;R,M;L,Q) model

For this model, Taylor and Templeton [1976] worked on a special case, R=N-1. They got the distribution of the number of busy servers and the unconditional expected waiting time for low priority customer. Here, we try to solve the general model for the steady state probability and the conditional expected waiting time of each waiting low priority customer at a given state.

Steady State Probability Distribution

We will first find all the P(n,q) in terms of P(0,0) and then sum them all up to 1 to find P(0,0). We summarize the procedure of finding the P(n,q) in terms of P(0,0) below.

STEP 1) Find P(n,0), n=1,2,...,R, in terms of P(0,0).

STEP 2) Define a recursive sequence to find P(n,0), n=R+1,...,N in terms of P(0,0).

STEP 3) For q=1,2,...,M-1, do

  STEP 3.1) Find P(R,q) in terms of P(0,0).

  STEP 3.2) Define two recursive sequences to find P(n,q), n=R+1,...,N in terms of P(0,0).

STEP 4) For q=M, do

  STEP 4.1) Find P(R,M) in terms of P(0,0).

  STEP 4.2) Define two different recursive sequences to find P(n,M), n=R+1,...,N in terms of P(0,0).

STEP 5) Find P(N,q), q=M+1,M+2,... in terms of P(0,0).

STEP 6) $\sum_{n,q} P(n,q)=1$ to find P(0,0).

Now we illustrate briefly each step below.

STEP 1) Find P(n,0), n=1,...,R

$$P(n,0) = \frac{\rho^n}{n!} P(0,0), \quad n=1,\ldots,R \tag{2.21}$$

STEP 2) Define a recursive sequence to find P(n,0), n=R+1,...,N. The balance equation at (N,0) is

$$(N\mu+\lambda 2)\ P(N,0) = \lambda_1 P(N-1,0)$$

Divide both sides by μ, we have

$$P(N-1,0) = [(N+\rho_2)/\rho_1]\ P(N,0)$$

$$\equiv A_{N-1}\ P(N,0) \tag{2.22}$$

where we have defined $A_N=1$, $A_{N-1}=(N+\rho_2)/\rho_1$.

Form the balance equation at state (N-1,0),(N-2,0),...(R+1,0) successively. We will have a recursive sequence

$$A_n = [(n+1+\rho)\ A_{n+1} - (n+2)\ A_{n+2}]/\rho_1\ , \quad n=N-2,N-3,\ldots,R \tag{2.23}$$

Then $$P(n,0) = (A_n/A_R)\ P(R,0)\ , \quad n=R+1,\ldots,N \tag{2.24}$$

Since P(R,0) is already defined in terms of P(0,0) in (2.21), so is P(n,0), n=R+1,...,N.

STEP 3) For q=1,2,...M-1, do

  STEP 3.1) Find P(R,q)

$$P(R,q) = \frac{\rho_2}{R}\left[\sum_{n=R}^{N} P(n,q-1)\right] \tag{2.25}$$

  STEP 3.2) Define two recursive sequences to find P(n,q), n=R+1,...,N.

Form the balance equation at state (N,q),(N-1,q),...,(R+1,q) successively, we will have two recursive sequences

$$A_n = [(n+1+\rho)A_{n+1} - (n+2)A_{n+2}]/\rho_1$$

$$B_n = [(n+1+\rho)B_{n+1} - (n+2)B_{n+2} - \rho_2 P(n+1,q-1)]/\rho_1, \quad n=N-2,N-3\ldots,R$$

with $A_N=1$, $A_{N-1} = (N+\rho_2)/\rho_1$

$$B_N=0 \ , \quad B_{N-1} = -\rho_2 P(N,q-1)/\rho_1 \tag{2.26}$$

Then we have

$$P(n,q) = (A_n/A_R)[P(R,q) - B_R] + B_n \ , \quad n=R+1,\ldots,N \tag{2.27}$$

STEP 4) q=M, do

STEP 4.1) Find P(R,M)

$$P(R,M) = \frac{\rho_2}{R}\left[\sum_{n=R}^{N} P(n,M-1)\right] \tag{2.28}$$

STEP 4.2) Define two recursive sequences to find P(n,M), n=R+1,...,N.

Form the blanace equation at state (N,M),(N-1,M),...,(R+1,M) successively, we will have two recursive sequences

$$C_n = [(n+1+\rho)C_{n+1} - (n+2)C_{n+2}]/\rho$$

$$D_n = [(n+1+\rho)D_{n+1} - (n+2)D_{n+2} - \rho_2 P(n+1,M-1)]/\rho \ , \quad n+N-2,\ldots,R$$

with $C_N=1$, $C_{N-1} = (N+\rho_2)/\rho$

$$D_N=0 \ , \quad D_{N-1} = -\rho_2 P(N,M-1)/\rho \tag{2.29}$$

Then we have

$$P(n,M) = (C_n/C_R)[P(R,M) - D_R] + D_n \ , \quad n=R+1,\ldots,N \tag{2.30}$$

STEP 5) Find P(N,q), q=M+1,...

$$P(N,q) = \left(\frac{\rho_2}{N}\right)^{q-M} P(N,M) \ , \quad q=M+1,\ldots \tag{2.31}$$

P(n,q), $q \geq M+1$, is a geometrical sequence so we can find the sum, $\sum_{q=M+1}^{\infty} P(N,q)$, without difficulty.

STEP 6) Find P(0,0)

$$1 = \sum_{n,q} P(n,q) = sP(0,0)$$

Then

$$P(0,0) = 1/s \tag{2.32}$$

Exhibit 2.5 is the block diagram of the above procedure.

As we can see from the block diagram, the number of operations is of the order of O(M(N-R+1)). So, it is very efficient. When R=N-1, the probabilities produced by our algorithm are exactly the same as Taylor and Templeton's.

The Conditional Expected Waiting Time for High Priority Customer

Since there is no high priority customer waiting in the queue, the conditional expected waiting time for high priority customer is 0 at any state.

The Conditional Expected Waiting Times for Low Priority Customer

Let $EW_M(n,q;k)$ denote the expected waiting time of the k^{th} low priority customer at state (n,q) for model D(N;R,M;L,Q), where $k \leq q$. Note that we have dropped the superscript ℓ in the notation without confusion. In this section we will derive a recursive formula to compute $EW_M(n,q;k)$ for each k at all

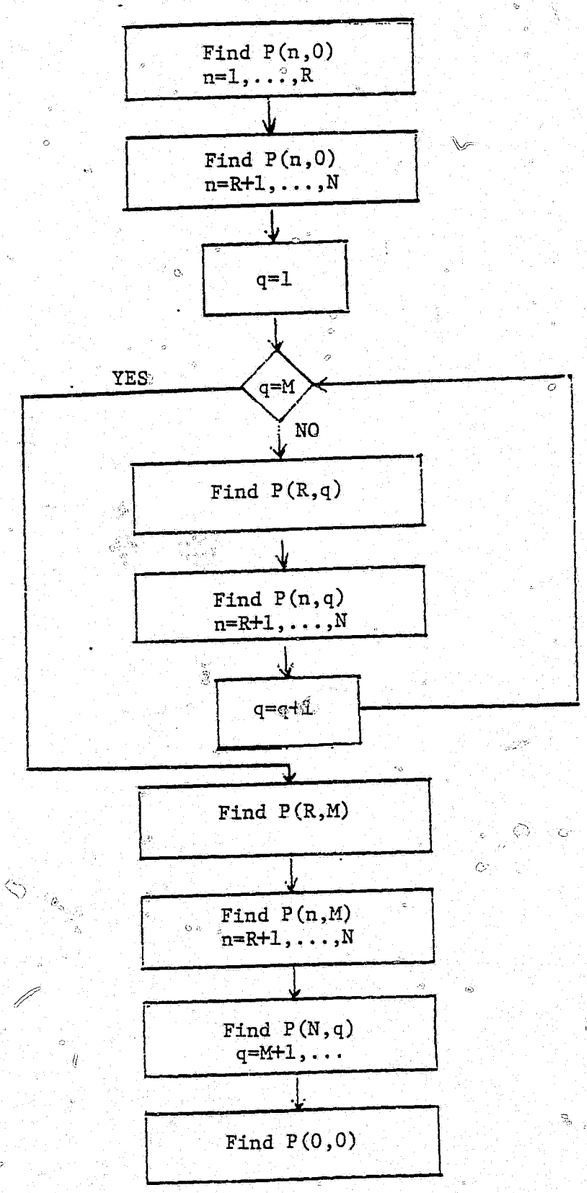


Exhibit 2.5 Block diagram of finding the P(0,0) in D(N;R,M;L,Q) model

feasible states (n,q). Following that we will carry out an example to show how it works. For R=N-1, Taylor and Templeton [1976] got the unconditional expected waiting time for this model. We will give a formula to unconditionalize our conditional expected waiting times and show numerically that they are identical. Then, we give an upper bound and a lower bound for $EW_M(n,q;k)$. Following that we will discuss a property about $EW_M(n,q;k)$ for this model. Finally, we examine an example obtained from the police department.

Now, we start to derive a recursive formula to compute $EW_M(n,q;k)$, $q \geq k \geq 1$. Because the restriction on the queue length, $EW_M(n,q;k)$ will depend on q as well as on n and k. It is very easy to see this at state (N,M) and state (N,M+1). At state (N,M+1), the first waiting low priority customer in the queue will start the service as soon as a server completes a service and returns to free. But at state (N,M), when a server completes a service and returns to free the system changes state to (N-1,M) and the first waiting low priority customer still has to wait in the queue. Hence, equation (2.14) will not hold in general for this model. We have to find a state such that the expected waiting time at that state is known to start the recursive process for each k. Fortunately, it is very easy to find such a state. For example, k=1, $EW_M(N,M+1;1) = 1/N\mu$. Because at state (N,M+1), there are M+1 low priority customer waiting in the queue and all N servers are busy. The first low priority customer will start the service as soon as a server completes the service. Since N servers are working with service rate μ per server, the expected waiting time for the first waiting low priority customer is $1/N\mu$. In general, $EW_M(N,M+k;k)=k/N\mu$, k=1,2,... Now, we start to compute $EW_M(N,q;k)$ for k=1. We have

$$EW_M(N,q;1) = 1/N\mu \quad , \quad q \geq M+1 \quad . \tag{2.33}$$

At state (N,M), we have

$$EW_M(N,M;1) = \frac{1}{N\mu+\lambda 2} + \frac{N\mu}{N\mu+\lambda 2}\, EW_M(N-1,M;1) + \frac{\lambda_2}{N\mu+\lambda 2}\, EW_M(N,M+1;1) \qquad (2.34)$$

The reason for (2.34) is the same as for (2.15). Multiply both sides by $N\mu+\lambda 2$ and substitute (2.33) into (2.34) we have

$$EW_M(N,M;1) = \frac{1+\lambda_2/N\mu}{N\mu+\lambda 2} + \frac{N\mu}{N\mu+\lambda 2}\, EW_M(N-1,M;1)$$

$$\equiv A_N + B_N\, EW_M(N-1,M;1) \qquad (2.35)$$

where we have defined $A_N=(1+\lambda_2/N\mu)/(N\mu+\lambda 2)$ and $B_N=N\mu/(N\mu+\lambda 2)$. By doing the same analysis at state $(N-1,M),\ldots,(R,M)$ successively, we will have two recursive sequences A_n and B_n

$$A_n = \frac{1+\lambda_1 A_{n+1}}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2}$$

$$B_n = \frac{n\mu}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2}, \quad n=N-1,\ldots,R \qquad (2.36)$$

Then we have

$$EW_M(R,M;1) = A_R$$

$$EW_M(n,M;1) = A_n + B_n\, EW_M(n-1,M;1), \quad n=R+1,\ldots,N \qquad (2.37)$$

Equations (2.33) and (2.37) have defined the conditional expected waiting times of the first low priority customer at those states (n,q) with $q \geq M$.

For $q=M-1$, follow the same analysis we will have two recursive sequences, A_n and B_n defined as

$$A_n = \frac{1+\lambda_1 A_{n+1}+\lambda_2 EW_M(n,M;1)}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2}$$

$$B_n = \frac{n\mu}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2}, \quad n=N-1,\ldots,R$$

with $$A_N = \frac{1+\lambda_2 EW_M(N,M;1)}{N+2}$$

$$B_N = \frac{N\mu}{N\mu+\lambda 2} \qquad (2.38)$$

Then

$$EW_M(R,M-1;1) = A_R$$

$$EW_M(n,M-1;1) = A_n + B_n\, EW_M(n-1,M-1;1), \quad n=R+1,\ldots,N \qquad (2.39)$$

In general, for $q=M,M-1,\ldots,1$, we can define A_n and B_n in the following way

$$A_n = \frac{1+\lambda_1 A_{n+1}+\lambda_2 EW_M(n,q+1;1)}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2}$$

$$B_n = \frac{n\mu}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2}, \quad n=N-1,\ldots,R \qquad (2.40)$$

with $$A_N = \frac{1+\lambda_2 EW_M(N,q+1;1)}{N\mu+\lambda 2}$$

$$B_N = \frac{N\mu}{N\mu+\lambda 2}$$

Then the conditional expected waiting time of the first low priority customer is

$$EW_M(R,q;1) = A_R$$

$$EW_M(n,q;1) = A_n + B_n\, EW_M(n-1,q;1), \quad n=R+1,\ldots, \qquad (2.41)$$

Note that equations (2.33) and (2.40) will completely define the expected waiting times of the first low priority customer at every feasible state. Exhibit 2.6 is the block diagram of finding the $EW_M(n,q;1)$ for n=R,...,N and q=M,M-1,...,1. As we can see from the block diagram, the number of operations to find the expected waiting times of the first low priority is of the order of O(M(N-R+1)). This algorithm can be extended to the k^{th} low priority with $k \geq 1$. Now we give an algorithm to compute $EW_M(n,q;k)$ for k=1,2,...,K at all feasible states.

STEP 1) [Assign the expected waiting times to 0 for all infeasible states]

$EW_M(n,m;0) \leftarrow 0$, n=R,...,N and m=0,1,...,M

$EW_M(n,M+1;k) \leftarrow 0$, n=R,...,N-1 and k=1,...,K

STEP 2) [Compute the expected waiting times for the k^{th} low priority customer]

$k \leftarrow 1$

STEP 3) [Assign the expected waiting time of the k^{th} low priority customer at the starting state]

$EW_M(N,m;k) \leftarrow k/N\mu$, m=M+k,M+k+1,...,M+K

STEP 4) $q \leftarrow M+k-1$

STEP 5) [Compute $EW_M(N,q;k)$ for q>M]

If q>M and $q \geq k$, then

$$EW_M(N,q;k) \leftarrow \frac{1}{N\mu+\lambda 2} + \frac{N\mu}{N\mu+\lambda 2} EW_M(N,q-1;k-1) + \frac{\lambda_2}{N\mu+\lambda 2} EW_M(N,q+1;k)$$

$q \leftarrow q-1$ and repeat this STEP.

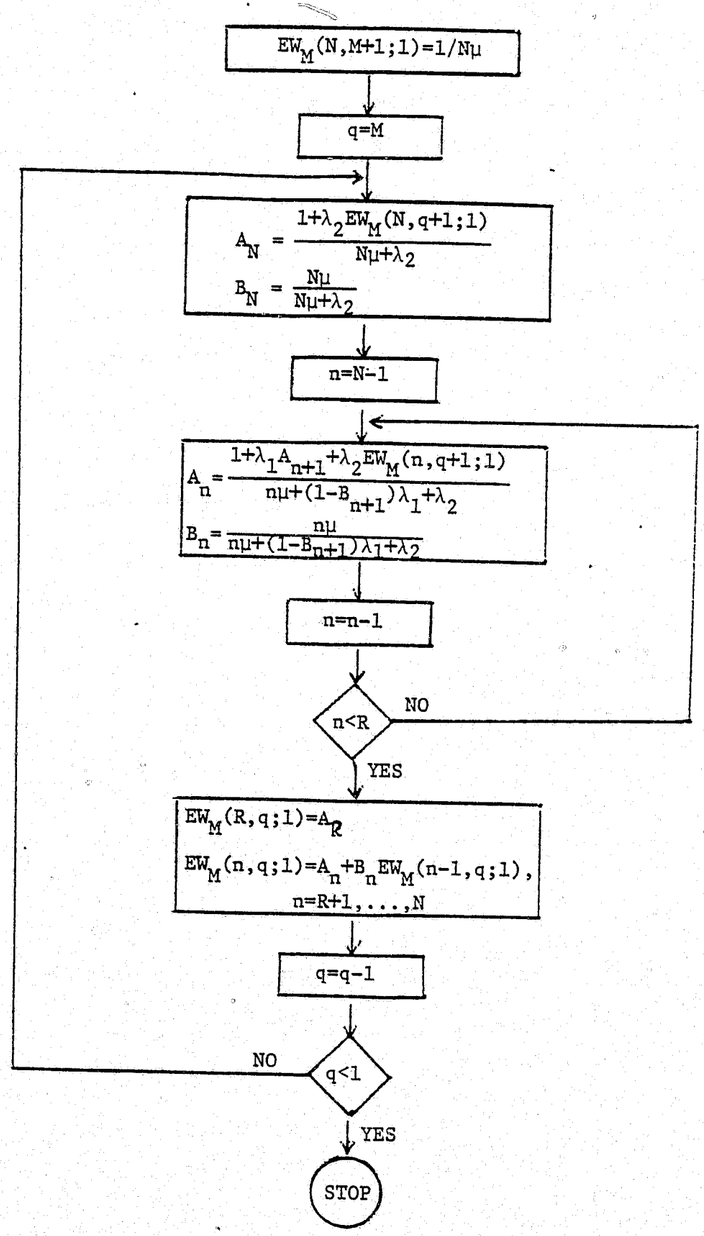


Exhibit 2.6 Block diagram of finding the expected waiting time for the 1st low priority customer in D(N;R,M;L,Q) model

If q>M and q<k, GO TO STEP 7)

STEP 6) [Compute $EW_M(n,q;k)$ for $q \leq M$]

$$A_N \leftarrow \frac{1+\lambda_2 EW_M(N,q+1;k)}{N\mu+\lambda 2}$$

$$B_N \leftarrow \frac{N\mu}{N\mu+\lambda 2}$$

STEP 6.1) Define A_n and B_n for n=N-1,...,R+1

$$A_n \leftarrow \frac{1+\lambda_1 A_{n+1}+\lambda_2 EW_M(n+1,M;k-1)}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2} \quad \text{when } q=M$$

$$B_n \leftarrow \frac{1+\lambda_1 A_{n+1}+\lambda_2 EW_M(n,q+1;k)}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2} \quad \text{when } q<M$$

and

$$B_n \leftarrow \frac{n\mu}{n\mu+(1-B_{n+1})\lambda_1+\lambda_2}$$

STEP 6.2) Define A_R

$$A_R \leftarrow \frac{1+R\mu EW_M(R,q-1;k-1)+\lambda_1 A_{R+1}+\lambda_2 EW_M(R+1,M;k-1)}{R\mu+(1-B_{R+1})\lambda_1+\lambda_2} \quad \text{when } q=M$$

$$A_R \leftarrow \frac{1+R\mu EW_M(R,q-1;k-1)+\lambda_1 A_{R+1}+\lambda_2 EW_M(R,q+1;k)}{R\mu+(1-B_{R+1})\lambda_1+\lambda_2} \quad \text{when } q<M$$

STEP 6.3) $EW_M(R,q;k) \leftarrow A_R$

$$EW_M(n,q;k) = A_n+B_n EW_M(n-1,q;k) \ , \quad n=R+1,\ldots,N$$

$q \leftarrow q-1$

If $q \geq k$, repeat STEP 6) o.w. GO TO STEP 7).

STEP 7) $k \leftarrow k+1$

If $k \leq K$, GO TO STEP 3) o.w. STOP.

The number of operations to find the conditional expected waiting time of the K^{th} low priority customer is of the order of $O(KM(N-R+1))$ for $K \leq M$ and is of the order of $O(M^2(N-R+1)$ for K>M. We do an example to show how this algorithm works below.

Example 2.2. Find the conditional expected waiting times for the first three low priority customers in D(5;3,2;L,Q) model with $\lambda_1=\lambda_2=\mu=1$. Exhibit 2.7 is the transition diagram of D(5;3,2;L,Q) model.

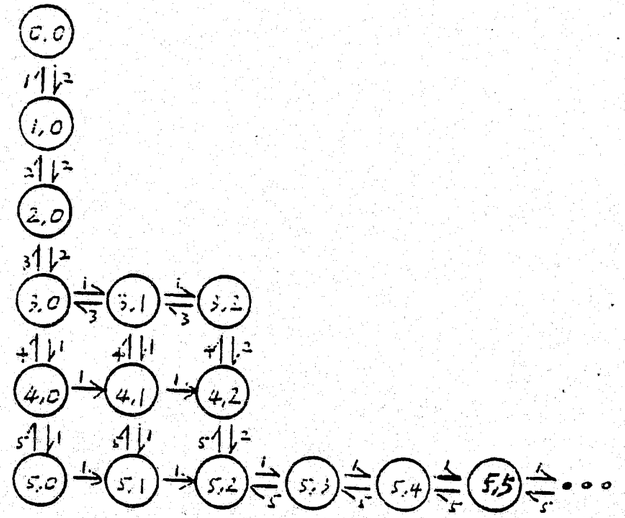


Exhibit 2.7 Transition diagram for D(5;3,2;L,Q)

In this example we have N=5, R=3, M=2, K=3, $\lambda_1=\lambda_2=\mu=1$.

STEP 1) $EW_2(n,m;0) = 0$, $n = 3,4,5$, $m = 0,1,2$

$EW_2(n,3;k) = 0$, $n = 3,4$, $k = 1,2,3$

STEP 2) $k = 1$

STEP 3) $EW_2(5,m;1) = .2$, $m = 3,4,5$

STEP 4) $q = 2$

STEP 5) Skip this step because of $q \leq M$

STEP 6) $A_5 = .2$, $B_5 = .833$

$A_4 = .232$, $B_4 = .774$

$A_3 = .292$

$EW_2(3,2;1) = .292$

$EW_2(4,2;1) = .458$

$EW_2(5,2;1) = .582$

$q = 2-1 = 1 \geq k = 1$, hence repeat this step.

STEP 6) $A_5 = .264$, $B_5 = .833$

$A_4 = .333$, $B_4 = .774$

$A_3 = .385$

$EW_2(3,1;1) = .385$

$EW_2(4,1;1) = .631$

$EW_2(5,1;1) = .789$

$q = 1-1 = 0 < k = 1$, hence GO TO STEP 7)

STEP 7) $k = 1+1 = 2 \leq K = 3$, hence GO TO STEP 3)

STEP 3) $EW_2(5,4;2) = EW_2(5,5;2) = .4$

STEP 4) $q = 2+2-1 = 3$

STEP 5) $EW_2(5,3;2) = .718$

$q = 3-1 = 2 = M$, GO TO STEP 6)

STEP 6) $A_5 = .286$, $B_5 = .833$

$A_4 = .362$, $B_4 = .774$

$A_3 = .704$

$EW_2(3,2;2) = .704$

$EW_2(4,2;2) = .906$

$EW_2(5,2;2) = 1.042$

$q = 2-1 = 1 < k = 2$, GO TO STEP 7)

STEP 7) $k = 2+1 = 3 \leq K = 3$, GO TO STEP 3)

STEP 3) $EW_2(5,5;3) = .6$

STEP 4) $q = 2+3-1 = 4$

STEP 5) $EW_2(5,4;3) = .8651$

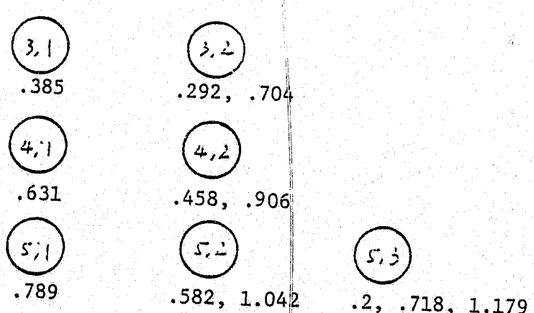
$q = 4-1 = 3 > M = 2$, repeat this step

STEP 5) $EW_2(5,3;3) = 1.179$

$q = 3-1 = 2 < k = 3$, GO TO STEP 7)

STEP 7) $k = 3+1 = 4 > K = 3$, STOP.

We summarize the results in the following diagram. The numbers under each state are the expected waiting times of each low priority customer at that state.



We are not surprised to see that $EW_2(3,2;1)<EW_2(4,2;1)<EW_2(5,2;1)$. We can explain this in the following way. At state (3,2) the 1^{st} low priority customer will start the service only when 1) a server completes the service or 2) another low priority customer arrives. But at state (4,2), the 1^{st} low priority customer will start the service only when 1) 2 servers complete the service or 2) another low priority customer arrives. Hence, the former is less than the latter. This relation holds in general. That is,

$$EW_M(R,q;k) < EW_M(R+1,q;k) < .... < EW_M(N,q;k) \qquad (2.41)$$

for $N>R$ and $q\geq k\geq 1$.

We also observe that $EW_2(n,2;1)<EW_2(n,1;1)$, $n=3,4,5$. We also can explain this in the following way: At state (3,2) the 1^{st} low priority customer will start the service only when 1) a server completes the service or 2) a low priority customer arrives. But at state (3,1), the 1^{st} low priority customer will start the service only when 1) a server completes the service or 2) two low priority customer arrive. Hence, the former is less than the latter. This relation also holds in general, that is

$$EW_M(n,M;k) < EW_M(n,M-1;k) < \ldots < EW_M(n,k;k) \qquad (2.42)$$

for $n=R,\ldots,N$ and $1\leq k<M$.

With R=N-1, Taylor and Templeton obtained the unconditional expected waiting time for low priority customer. We would like to check our result with theirs when R=N-1. In order to make the comparison, we have to unconditionalize our conditional expected waiting time. To unconditionalize the conditional expected waiting times we have to weigh the conditional expected waiting times by the corresponding steady state probabilities. Let EW_M denote the unconditional expected waiting time of low priority in D(N;R,M;L,Q) model. Then

$$EW_M = \sum_{q=0}^{M-1} \sum_{n=R}^{N} P(n,q)\ EW_M(n,q+1;q+1) + \sum_{n=R}^{N-1} P(n,M)\ EW_M(n+1,M;M)$$

$$+ \sum_{q=M}^{\infty} P(N,q)\ EW_M(N,q+1;q+1) \qquad (2.43)$$

The reason for (2.43) is, when a low priority customer arrives and finds that 1) the system is at state (n,q), $n=R,\ldots,N$ and $q=0,\ldots,M-1$, then he/she has to wait in the queue at the $(q+1)^{st}$ position with the expected waiting time equals to $EW_M(n,q+1;q+1)$, 2) the system is at state (n,M), $n=R,\ldots,N-1$, then he/she has to wait in the queue at the M^{th} position with the expected waiting time equals to $EW_M(n+1,M;M)$, 3) the system is at state (N,q), $q\geq M$, then he/she has to wait in the queue at the $(q+1)^{st}$ position with the expected waiting time equals to $EW_M(N,q+1;q+1)$. This algorithm has been programmed in FORTRAN.

In the program, it computes $P(n,q)$, $EW_M(n,q;k)$ and EW_M. In computing the EW_M, it includes the first two terms of (2.43) and part of the third term (only up to q=20). Exhibit 2.8 is a list of comparisons of the expected waiting times of low priority customer in D(N;R,M;L,Q) model between the theoretical value and the result from the program for R=N-1. As expected, our results are not greater than the theoretical values and all of them are very close to the theoretical values.

Now, we start to discuss the bounds. As we mentioned earlier in Section 2.2, the expected waiting time of low priority customer in $D(N;R,\infty;L,Q)$ model can be used as an upper bound in the D(N;R,M;L,Q) model. That is, $EW_M(n,q;k)<EW_\infty(n,q;k)$ for any finite M at any corresponding state (n,q) with $q\geq k$. Furthermore, $EW_M(n,q;k)$ is an increasing function of M. That is,

$$EW_M(n,q;k) < EW_{M+1}(n,q;k) < EW_\infty(n,q;k) \qquad (2.44)$$

for any $M\geq 1$ at any corresponding state (n,q).

N	M	λ_1	λ_2	μ	Expected Waiting Time	
					Theoretical Value	Our Result
5	1	1	2	1	.135	.135
5	1	2	1	1	.1282	.1282
5	5	1	2	1	.3029	.3029
5	5	2	1	1	.2199	.2199
5	10	1	2	1	.3552	.3552
5	10	2	1	1	.2246	.2246
10	1	2	6	1	.09932	.09927
10	1	6	2	1	.06412	.06412
10	5	2	6	1	.2182	.2181
10	5	6	2	1	.1114	.1114
10	10	2	6	1	.3158	.3152
10	10	6	2	1	.1142	.1142
15	1	.05	.25	.034	.2278	.2278
15	5	.05	.25	.034	.3979	.3979
15	10	.05	.25	.034	.4358	.4357

Exhibit 2.8 Comparisons of the unconditional expected waiting time in D(N;N-1,M;L,Q) model

We can explain (2.44) in this way: The smaller the M, the larger "pressure" the system has. The larger pressure the system has, the "faster" the servers will work. The faster the servers work, the less conditional expected waiting time of low priority customer will have. For example, (R,1) is a feasible state for all of this model. When M=1, state (R,1) is at the limit. One more low priority customer arrives will begin the service of "the first low priority". When M=5, state (R,1) is not at the limit. It can tolerate four more low priority arrivals. When M=∞ the system will not have any pressure at all. Hence $EW_\infty(n,q;k)$ is an upper bound of $EW_M(n,q;k)$ and it does not take

too long to reach the upper bound. For example, at state (12,1) in D(15;12,M;L,Q) model with λ_1=.05, λ_2=.25, μ=.034, we have

M	1	2	3	4	5	6	7	∞
$EW_M(12,1;1)$	1.59	2.25	2.53	2.65	2.71	2.74	2.75	2.76

when M=7, $EW_7(12,1;1)$ is very close to $EW_\infty(12,1;1)$. So, for any fixed M, $EW_M(n,q;k)$ is bounded above by $EW_\infty(n,q;k)$ which is equal to $EW_\infty(n,k;k)$. And $EW_\infty(n,k;k)$ can be found very easily by the method in Section 2.2. Now, we try to find a lower bound of $EW_M(n,q;k)$. From equation (2.42), $EW_M(n,M;k)$ is a lower bound of $EW_M(n,q;k)$,

$$EW_M(n,M;k) \le EW_M(n,q;k) \le EW_M(n,k;k) < EW_\infty(n,k,k) \qquad (2.45)$$

for n=R,...,N and $1 \le k \le q \le M$.

Hence, $EW_M(n,q;k)$ is bounded below by $EW_M(n,M;k)$ for all $k \le q \le M$ and the lower bound is exact when q=M. To find $EW_M(n,M;k)$ we have to carry out the recursive sequences A_n and B_n which are defined in this section. For q<M, it will be easier to find the lower bound than to find the actual value of $EW_M(n,q;k)$. We do a simple example to illustrate how to find the bounds.

Example 2.3. A low priority customer arrives and finds the system at state (3,0), that is, 3 servers are busy and no low priority customer in the queue. The system has 4 servers in total and uses the dispatching procedure D(4;3,3;L,Q) to dispatch the servers. Hence, the arriving low priority customer will have to wait in the queue for his turn for service. The input rates to the system are λ_1=1, λ_2=1 and the service rate per server is μ=1. What is the maximum expected waiting time and what is the minimum expected waiting time for this

low priority customer? In this example, we have N=4, R=3, M=3, $\lambda_1=\lambda_2=\mu=1$. The system is at state (3,1) and "the low priority" is in the first position in the queue. So, the conditional expected waiting time is $EW_3(3,1;1)$. From above, we have

$EW_3(3,3;1) < EW_3(3,1;1) < EW_\infty(3,1;1)$

(1) Compute $EW_\infty(3,1;1)$, the upper bound.

From Section 2.2, we have

$$D_N = \frac{1}{N\mu}$$

$$D_{N-1} = D_R = \frac{1+\lambda_1 D_N}{R\mu} = \frac{1+\lambda_1/N\mu}{R\mu} = \frac{N\mu+\lambda_1}{RN\mu^2} = .4167$$

Hence, we have

$$EW_\infty(3,1;1) = .4167$$

(2) Compute $EW_3(3,3;1)$, the lower bound.

From this Section, we have

$$A_N = \frac{1+\lambda_2/N\mu}{N\mu+\lambda_2} \qquad B_N = \frac{N\mu}{N\mu+\lambda_2}$$

$$A_{N-1} = A_R = \frac{1+\lambda_1 A_N}{R\mu+(1-B_N)\lambda_1+\lambda_2}$$

$$= \frac{(N\mu+\lambda_2)N\mu+N\mu\lambda_1+\lambda_1\lambda_2}{(R\mu+\lambda_2)(N\mu+\lambda_2)N\mu+N\mu\lambda_1\lambda_2} = .2976$$

Hence,

$$EW_3(3,3;1) = .2976$$

That is,

$$.2976 < EW_3(3,1;1) < .4167 \qquad \text{Q.E.D.}$$

In general, given D(N;R,M;L,Q), it is not hard to find the upper bound analytically for $EW_M(n,q;k)$, but it is very tedious (but not impossible) to find the lower bound analytically, especially for k>1. But it should be very easy to compute the numerical value of the lower bound for any k.

Now, let's define the boundary state for model D(N;R,M;L,Q). The boundary states in D(N;R,M;L,Q) model are those states which have exact M low priority customer in the queue. For example, in D(5;3,1;L,Q), the boundary states are (3,1), (4,1) and (5,1), in D(5;3,2;L,Q), the boundary states are (3,2), (4,2) and (5,2). Exhibits 2.9 and 2.10 are the configurations of the boundary state (3,1) and the boundary state (3,2) in D(5;3,1;L,Q) and D(5;3,2;L,Q) respectively.

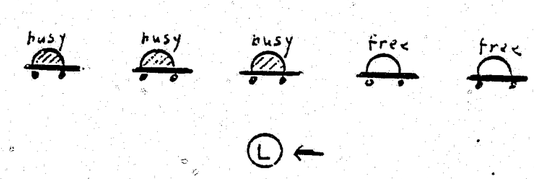


Exhibit 2.9 The configuration of boundary state (3,1) in D(5;3,1;L,Q)

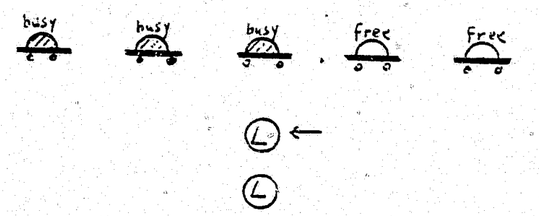


Exhibit 2.10 The configuration of boundary state (3,2) in D(5;3,2;L,Q)

In D(5;3,1;L,Q), at state (3,1), the first low priority customer (indicated by the arrow in Exhibit 2.9) is experiencing the same "forces" as the first low

priority customer (indicated by the arrow in Exhibit 2.10) at state (3,2) in D(5;3,2;L,Q). The forces are 1) server completes the service and returns to free, 2) high priority customer arrives, and 3) low priority customer arrives. Those are the only forces can make the system change the state. Since they are experiencing the same forces, they should have the same (expected) waiting time. That is, $EW_1(3,1;1)=EW_2(3,2;1)$. The same argument applies to all the boundary states in the model D(N;R,M;L,Q). Hence, in general, we have

$$EW_1(n,1;1) = EW_2(n,2;1) = \ldots = EW_M(n,M;1) \qquad (2.46)$$

for n=R,...,N.

Let's generalize the definition of the boundary state and define the m^{th} boundary states, $m \leq M$, in D(N;R,M;L,Q) are the states which have exactly (M-m+1) low priority customers in the queue. So, the first boundary states are the boundary states defined earlier. The 2^{nd} boundary states in D(5;3,3;L,Q) are (3,2), (4,2) and (5,2), the 3^{rd} boundary states in D(5;3,3;L,Q) are (3,1), (4,1) and (5,1). We can extend (2.46) to the m^{th} boundary states. That is,

$$EW_m(n,1;1) = EW_{m+1}(n,2;1) = \ldots = EW_M(n,M-m+1;1) \qquad (2.47)$$

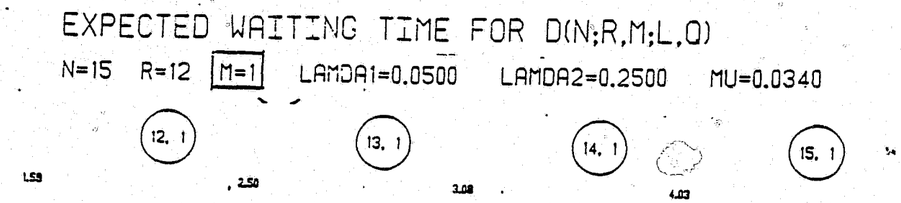
for n=R,...,N and $1 \leq m \leq M$.

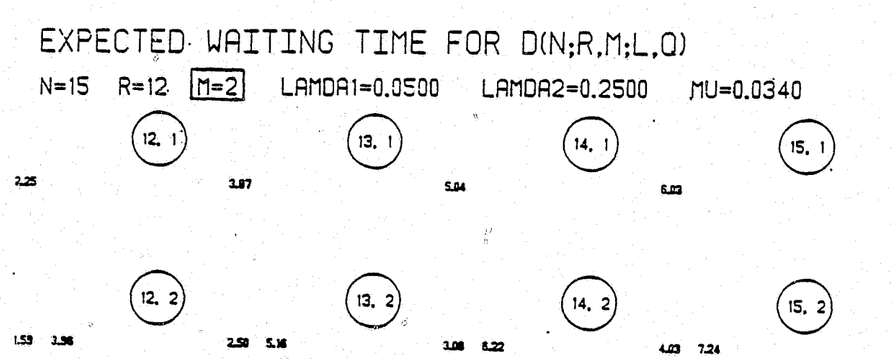
We can extend (2.47) further more to the k^{th} low priority. That is,

$$EW_m(n,k;k) = EW_{m+1}(n,k+1;k) = \ldots = EW_M(n,M-m+k;k) \qquad (2.48)$$

for n=R,...,N and $M \geq m \geq k \geq 1$.

Exhibit 2.11 is the diagram of the conditional expected waiting times for models D(15;12,1;L,Q), D(15;12,2;L,Q) and D(15;12,3;L,Q) with $\lambda_1=.05$, $\lambda_2=.25$ and $\mu=.034$. As we can see that





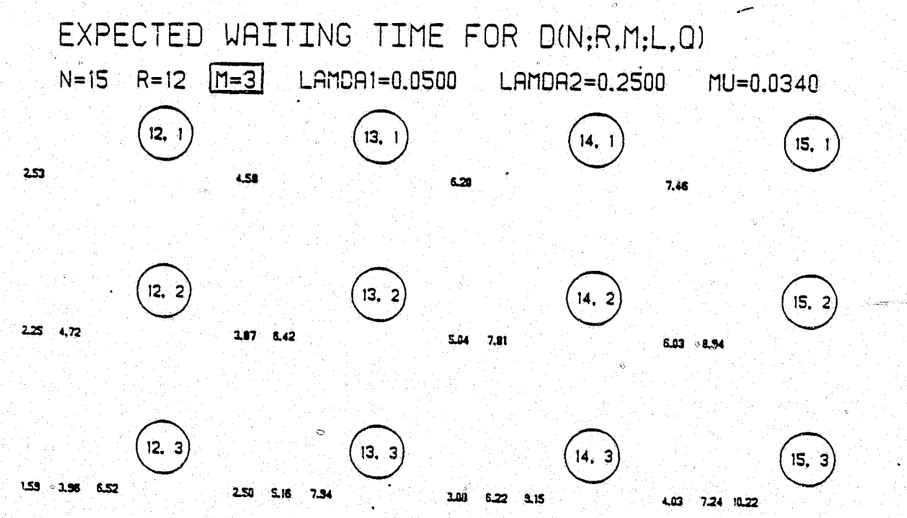


Exhibit 2.11 Comparisons of the expected waiting times with different M-cutoff in D(15;12,M;L,Q) models

$$EW_1(12,1;1) = EW_2(12,2;1) = EW_3(12,3;1) = 1.59$$

and $$EW_2(12,2;2) = EW_3(12,3;2) = 3.96$$

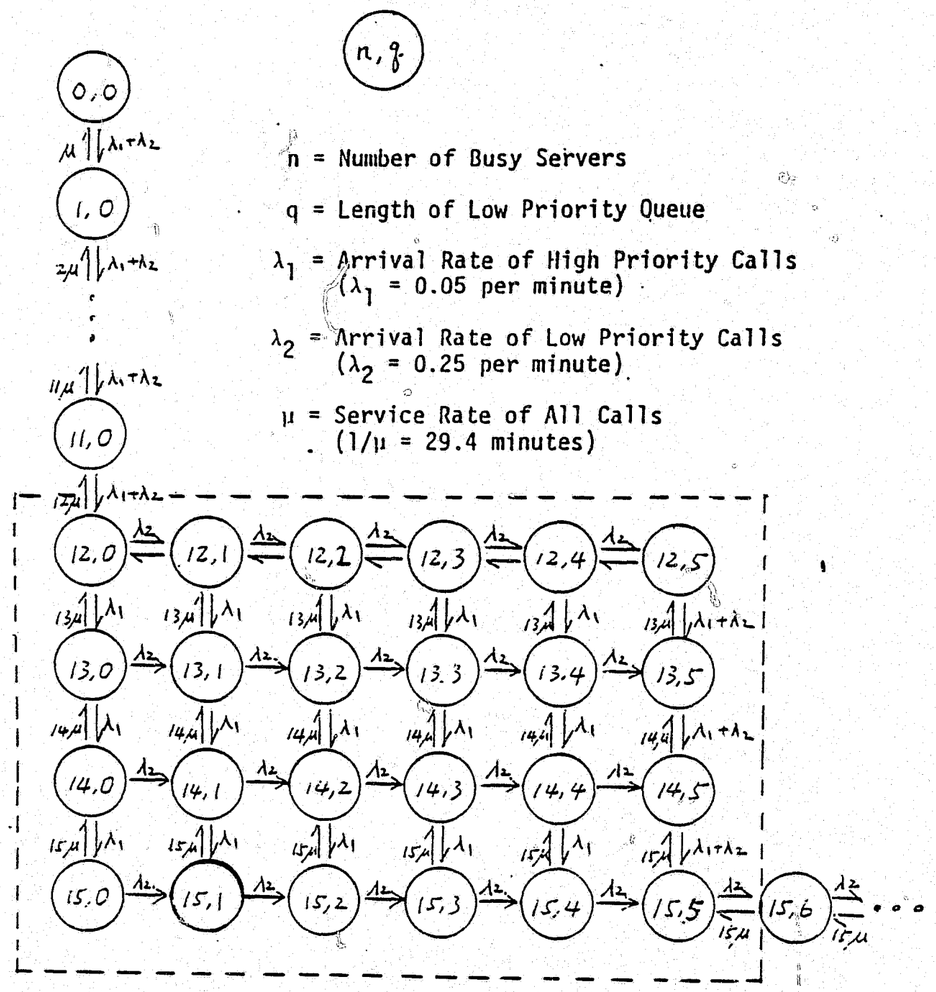
and etc. which confirm the relations in (2.48).

Following example is obtained from the police department. A police department has 15 cars on duty. The utilization factor is .6 and on average 17% of total calls are emergency (high priority). The expected service time is 30 min. for both types of calls. From this information we can compute the arrival rates and the service rate. That is, λ_1=3/hr=.05/min,λ_2=15/hr=.25/min and μ=2/hr=.034/min per car. The department head decides to reserve 3 cars for emergency calls and at the same time do not allow more than 5 nonemergency calls waiting in the queue if there is a car available. The emergency call will not be able to wait if a car is not available at its arrival time. This is D(15;12,5;L,Q) model. Apply our algorithm, we got the steady state probability distribution and the conditional expected waiting times for each low priority call at every state. Exhibit 2.12 is the transition diagram for D(15;12,5;L,Q) model. Exhibits 2.13 and 2.14 provide the steady state probability and the conditional expected waiting times, respectively, for those states that are in the dashed box in Exhibit 2.12.

In Exhibit 2.10, the number below each state is the steady state probability of that state. For example, at state (14,3), P(14,3) = .00034. In Exhibit 2.14, the values below each state are the expected waiting times for each low priority call in the queue. For example, at state (14,3), there are three low priority calls waiting in the queue and the expected waiting time for the 1st call is 6.2 min., for the 2nd call is 8.85 min., for the 3rd call is 11.54 min. at the instant when the system went into state (14,3). Several interesting results emerge from Exhibit 2.14. First, the more low priority

calls waiting, the less is the expected waiting time in the same position. Second, the busier the system, the greater the expected waiting time. We already explained these properties before. Third, if we were to choose M=4 instead of M=5, then state (12,4) would be a boundary state of the system. Consequently, if one more low priority call arrives then the first call in the queue would start the service and the system enters state (13,4). In comparing the expected waiting time for the fourth call at state (13,4), i.e., $EW_4(13,4;4)$ which is equal to $EW_5(13,5;4)$=10.71, to the expected waiting time for the fifth call at state (13,5), i.e., $EW_5(13,5;5)$=11.88, it should be noted that the former is less than the latter. The difference indicates the gain, i.e., less waiting time, for the low priority call, if we were to switch from M=5 to M=4. The trade-off here is it will cause more high priority calls lost.

In the next section we will discuss the D(N;R,M;Q,Q) model, that is, the high priority call will also be queued in the high priority queue if there is no free car available at the time it arrives.



n = Number of Busy Servers

q = Length of Low Priority Queue

λ_1 = Arrival Rate of High Priority Calls (λ_1 = 0.05 per minute)

λ_2 = Arrival Rate of Low Priority Calls (λ_2 = 0.25 per minute)

μ = Service Rate of All Calls ($1/\mu$ = 29.4 minutes)

Exhibit 2.12 Transition diagram for D(15;12,5;L,Q) model

STEADY STATE PROBABILITY FOR D(N;R,M;L,Q)

N=15 R=12 M=5 LAMDA1=0.0500 LAMDA2=0.2500 MU=0.0340

THE PROBABILITY OF HIGH PRIORITY CUSTOMER LOST IS 0.0065

THE PROBABILITY OF LOW PRIORITY CUSTOMER WAIT IS 0.1532

(12, 0)	(13, 0)	(14, 0)	(15, 0)
0.06488	0.00457	0.00031	0.00002
(12, 1)	(13, 1)	(14, 1)	(15, 1)
0.04275	0.00469	0.00042	0.00003
(12, 2)	(13, 2)	(14, 2)	(15, 2)
0.02935	0.00382	0.00041	0.00004
(12, 3)	(13, 3)	(14, 3)	(15, 3)
0.02050	0.00299	0.00034	0.00003
(12, 4)	(13, 4)	(14, 4)	(15, 4)
0.01463	0.00213	0.00027	0.00003
(12, 5)	(13, 5)	(14, 5)	(15, 5)
0.01045	0.00947	0.00549	0.00324

Exhibit 2.13 Steady state probability distribution for D(15;12,5;L,Q)

EXPECTED WAITING TIME FOR D(N;R,M;L,Q)

N=15 R=12 M=5 LAMDA1=0.0500 LAMDA2=0.2500 MU=0.0340

THE PROBABILITY OF HIGH PRIORITY CUSTOMER LOST IS 0.0065

THE PROBABILITY OF LOW PRIORITY CUSTOMER WAIT IS 0.1532

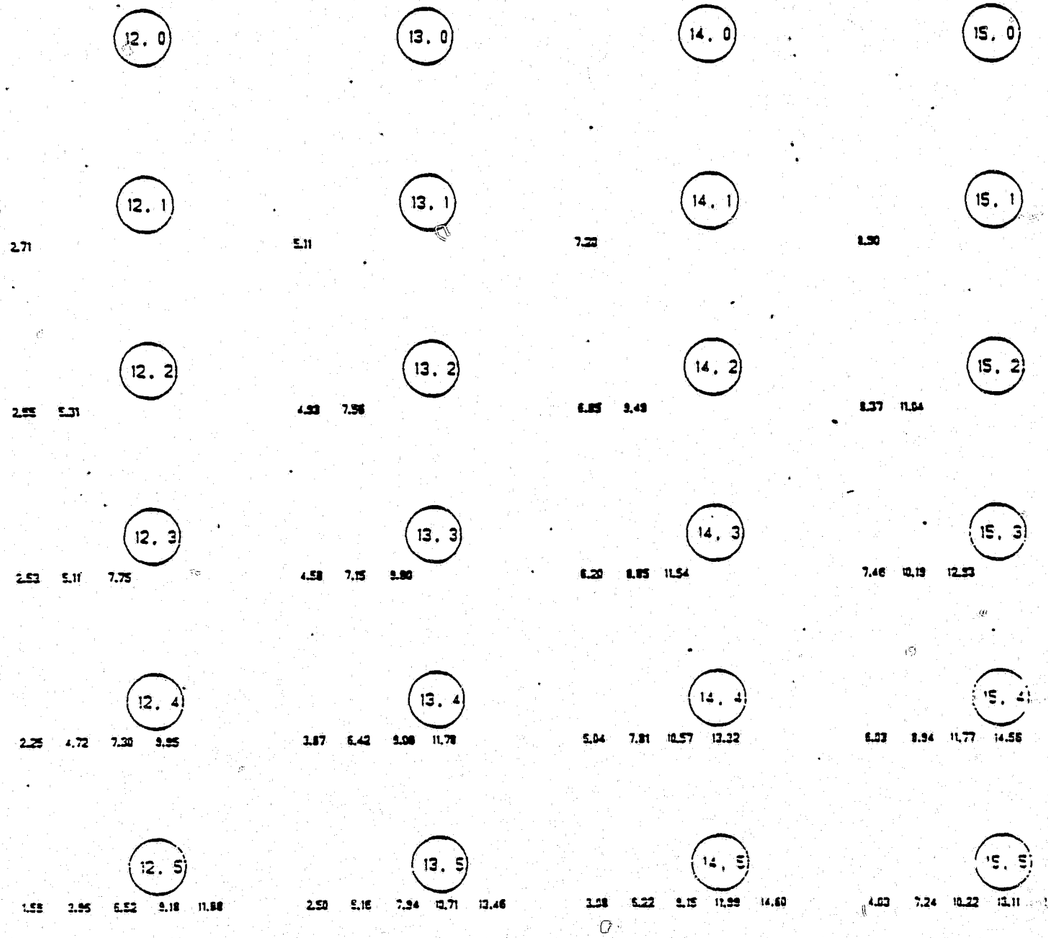


Exhibit 2.14 Conditional expected waiting times of low priority customers in D(15;12,5;L,Q)

3. D(N;R,M;Q,Q) Model

In this model there are two queues, the high priority queue and the low priority queue. Unlike the D(N;R,M;L,Q) model, the high priority customer will be put into the high priority queue if the system is full at the time he/she arrives. The low priority customer will be put into the low priority queue whenever service can not be provided immediately. Customers in the queue will be served in the FIFO order within the class. In this section we also assume that both high priority customer and low priority customer have independent Possion arrival time with the arrival rates λ_1 and λ_2, respectively. The server has exponential service time with the expected service time $1/\mu$ for both types of customers. Let $\rho_1=\lambda_1/\mu$, $\rho_2=\lambda_2/\mu$ and $\rho=\rho_1+\rho_2$.

We need three variables to characterize the state of the D(N;R,M;Q,Q) model. Let (n,q_1,q_2) designate the state in the D(N;R,M;Q,Q) model where n is the number of busy servers, q_1 is the number of customer in the high priority queue and q_2 is the number of customer in the low priority queue. As it is in the D(N;R,M;L,Q) model, we should consider following three cases (with the first two cases are special cases of the third one).

(1) R=N and M=∞.

(2) R<N and M=∞.

(3) R<N and M is finite.

In the following of this section we will study each case the steady state distribution and the conditional expected waiting times for both the high priority and low priority customers at a given state.

Let $P(n,q_1,q_2)$ denote the steady state probability at state (n,q_1,q_2), $EW_M^h(n,q_1,q_2;k)$, $k\le q_1$, denote the conditional expected waiting time of the k^{th} high priority customer at state (n,q_1,q_2) and $EW_M^\ell(n,q_1,q_2;k)$, $k\le q_2$, denote the

conditional expected waiting time of the k^{th} low priority customer at state (n,q_1,q_2) in D(N;R,M;Q,Q) model.

3.1 D(N;N,∞;Q,Q) Model

For this model no server is reserved for high priority customer. When the system is not full we will send a server to the arriving customer no matter what priority the customer is. If all the servers are busy at the time a customer arrives then he/she has to be queued in the corresponding queue. When a server completes service and returns to free it will first check the high priority queue. If the high priority queue is not empty then the first customer will start the service. If the high priority queue is empty then it will check the low priority queue. If the low priority queue is not empty then the first customer in the low priority queue will start the service. If the low priority queue is also empty then the server will remain free. Exhibit 3.1 is the transition diagram for the D(N;N,∞;Q,Q) model.

Steady State Probability Distribution

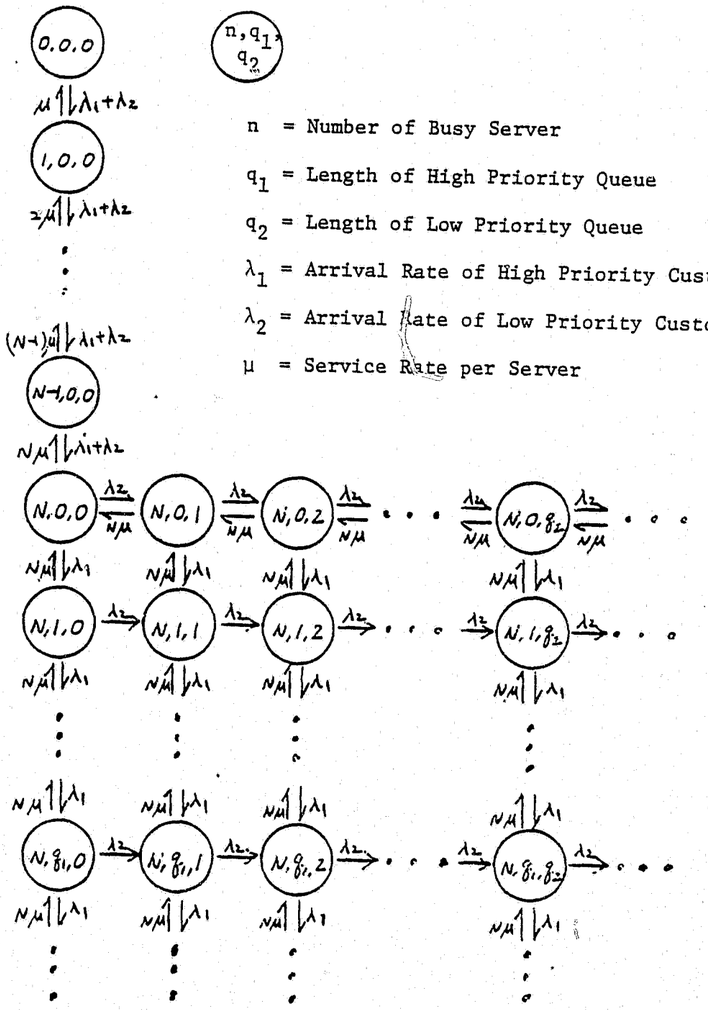
This is a special model of Taylor and Templeton's [1980] with R=N. In their work, they obtained the probability distribution of the number of busy servers. Hence, they got P(0,0,0), the probability of all the servers are free.

$$P(0,0,0) = \left[\sum_{n=0}^{N-1} \rho^n/n! + (\rho^N/(N-1)!)(N/(N-\rho_1))/(N-\rho_2 N/(N-\rho_1)) \right]^{-1} \qquad (3.1)$$

We will start to derive the steady state probability distribution by knowing the P(0,0,0). We summarize the steps of finding these steady state probabilities below.

STEP 1) Compute P(n,0,0), n=1,2,...,N.

STEP 2) Prove a theorem and compute $P(N,q_1,0)$, $q_1=1,2,\ldots,Q_1$.



n = Number of Busy Server

q_1 = Length of High Priority Queue

q_2 = Length of Low Priority Queue

λ_1 = Arrival Rate of High Priority Customer

λ_2 = Arrival Rate of Low Priority Customer

μ = Service Rate per Server

Exhibit 3.1 Transition Diagram for D(N;N,∞;Q,Q) Model

STEP 3) For each $q_2 \geq 1$, do

STEP 3.1) Compute $P(N,0,q_2)$

STEP 3.2) Compute $P(N,q_1,q_2)$, $q_1=1,2,\ldots,Q_1$ recursively.

Now, we outline these steps.

STEP 1) Compute $P(n,0,0)$, $n=1,2,\ldots,N$.

Before the system is full, it is an M/M/N queue. Hence

$$P(n,0,0) = \frac{\rho^n}{n!} P(0,0,0), \quad n=1,2,\ldots,N \tag{3.2}$$

STEP 2) Prove a theorem and compute $P(N,q_1,0)$, $q_1=1,2,\ldots,Q_1$.

THEOREM 3.1: In D(N;R,M;Q,Q) model, $P(N,q_1,0)=x^{q_1}P(N,0,0)$

where $x = \left[(N+\rho) - \sqrt{(N+\rho)^2-4N\rho_1} \right]/2N$

PROOF: We assume that the steady state exists. The balance equation at state $(N,q_1,0)$, $q_1=1,2,\ldots$ is

$$(N\mu+\lambda_1+\lambda_2)\, P(N,q_1,0) = \lambda_1 P(N,q_1-1,0) + N\mu P(N,q_1+1,0)$$

Divide both sides by μ and move all the terms to the RHS, we have

$$NP(N,q_1+1,0) - (N+\rho)P(N,q_1,0) + \rho_1 P(N,q_1-1,0) = 0 \tag{3.3}$$

All the coefficients in (3.3) are constant and (3.3) holds for all $q_1 \geq 1$. Hence, we have $P(N,q_1,0)=x^{q_1}P(N,0,0)$ and substitute this into (3.3), we have

$$Nx^2 - (N+\rho)x + \rho_1 = 0$$

$$x = \frac{(N+\rho) \pm \sqrt{(N+\rho)^2-4N\rho_1}}{2N}$$

Since the steady state exists, $0<x<1$.

Hence, $x = \left[(N+\rho) - \sqrt{(N+\rho)^2-4N\rho_1} \right]/2N$ Q.E.D.

Note that Theorem 3.1 holds for the general model, D(N;R,M;Q,Q), hence it holds for the special models also. For fixed ρ, x is an increasing function of ρ_1 and $x \to \rho/N$ when $\rho_1 \to \rho$. Using Theorem 3.1, we can compute $P(N,q_1,0)$, $q_1=1,2,\ldots$ From the computation point of view we can ignore those states which have probabilities smaller than ε, for example $\varepsilon=10^{-7}P(0,0,0)$. Let Q_1 be the integer such that $P(N,Q_1+1,0)<\varepsilon$ and $P(N,Q_1,0)\geq\varepsilon$. Then we can compute $P(N,q_1,0)$, $q_1=1,2,\ldots,Q_1$.

STEP 3) For $q_2 \geq 1$, do

STEP 3.1) Compute $P(N,0,q_2)$

$$P(N,0,q_2) = \left[\sum_{q_1=0}^{\infty} P(N,q_1,q_2-1) \right](\rho_2/N)$$

$$\cong \left[\sum_{q_1=0}^{Q_1} P(N,q_1,q_2-1) \right](\rho_2/N) \tag{3.4}$$

STEP 3.2) Compute $P(N,q_1,q_2)$, $q_1=1,2,\ldots,Q_1$ recursively.

By ignoring the flow from the state (N,Q_1+1,q_2) into the state (N,Q_1,q_2) and the flow from the state (N,Q_1,q_2) into the state (N,Q_1+1,q_2), the balance equation at state (N,Q_1,q_2) is

$$(N\mu+\lambda_2)P(N,Q_1,q_2) = \lambda_1 P(N,Q_1-1,q_2) + \lambda_2 P(N,Q_1,q_2-1)$$

Dividing both sides by μ, we can express $P(N,Q_1-1,q_2)$ in terms of $P(N,Q_1,q_2)$

$$P(N,Q_1-1,q_2) = \frac{N+\rho_2}{\rho_1} P(N,Q_1,q_2) - \frac{\rho_2}{\rho_1} P(N,Q_1,q_2-1)$$

$$\equiv C_{Q_1-1} P(N,Q_1,q_2) + D_{Q_1-1} \qquad (3.5)$$

where we have defined $C_{Q_1}=1$, $C_{Q_1-1}=\frac{N+\rho_2}{\rho_1}$, $D_{Q_1}=0$ and $D_{Q_1-1}=\frac{-\rho_2}{\rho_1} P(N,Q_1,q_2-1)$.

By doing the same analysis, we obtain two recursive sequences C_q and D_q, $q=Q_1-2,\ldots,0$

$$C_q = [(N+\rho)C_{q+1} - NC_{q+2}]/\rho_1$$

$$D_q = [(N+\rho)D_{q+1} - ND_{q+2} - \rho_2 P(N,q+1,q_2-1)]/\rho_1$$

Then we can express $P(N,q,q_2)$ in terms of $P(N,Q_1,q_2)$.

$$P(N,q,q_2) = C_q P(N,Q_1,q_2) + D_q, \quad q=0,1,\ldots,Q_1 \qquad (3.6)$$

For $q=0$, we have

$$P(N,0,q_2) = C_0 P(N,Q_1,q_2) + D_0$$

Since, we already know $P(N,0,q_2)$ from (3.4), we have

$$P(N,Q_1,q_2) = [P(N,0,q_2)-D_0]/C_0 \qquad (3.7)$$

Substitute (3.7) into (3.6) we will have $P(N,q_1,q_2)$, $q_1=1,\ldots,Q_1$. We can keep increasing q_2 by 1 and repeat STEP 3) until $P(N,0,q_2)$ is smaller than a pre-specified value. Note that the steady state can be reached only when $\rho<N$.

Conditional Waiting Time Distribution for High Priority Customer

Let W_k^h denote the time that the k^{th} high priority customer spent in the queue given that he/she has to wait. Then W_k^h is Erlang distributed with

parameters $N\mu$ and k. That is, the density function of W_k^h is

$$f_{W_k^h}(t) = \frac{(N\mu)^k}{(k-1)!} t^{k-1} e^{-N\mu t}, \quad k=1,2,\ldots \qquad (3.8)$$

The conditional expected waiting time of the k^{th} high priority customer, $EW_\infty^h(N,q_1,q_2;k)$, is

$$EW_\infty^h(N,q_1,q_2;k) = \frac{k}{N\mu} \qquad (3.9)$$

Note that the conditional (expected) waiting time of the k^{th} high priority customer is only dependent on the position in the queue, i.e., k, and not on the number of high priority customer in the queue and the number of low priority customer in the queue.

Conditional Waiting Time Distribution For Low Priority Customer

Let $W^\ell_{q_1+q_2+1}$ denote the time a low priority spent in the queue when he/she arrives and finds the system is at state (N,q_1,q_2). Then the density function of $W^\ell_{q_1+q_2+1}$ can be expressed in the transformed space (Davis, 1966).

$$f_{W^\ell_{q_1+q_2+1}}(s) = \int_0^\infty e^{-st} W^\ell_{q_1+q_2+1}(t)\, dt = \left[\frac{(s+N\mu+\lambda_1)-\sqrt{(s+N\mu+\lambda_1)^2-4N\mu_1}}{2\lambda_1}\right]^{q_1+q_2+1} \qquad (3.10)$$

where $q_1 \geq 0$, $q_2 \geq 0$.

The conditional expected waiting time of this low priority customer is

$$EW_\infty^\ell(N,q_1,q_2+1;q_2+1) = \frac{q_1+q_2+1}{N\mu-\lambda 1} \qquad (3.11)$$

Suppose, in $D(N;R,\infty;Q,Q)$ model with $R \leq N$, there are q_1 customer waiting in the high priority queue at the time the q_2^{th} low priority customer arrives.

We want to find the conditional expected waiting time of the k^{th} low priority customer, $k \leq q_2$, that is, the $EW_\infty^\ell(n,q_1,q_2;k)$. The conditional expected waiting time of the k^{th} low priority customer in D(N;R,∞;Q,Q) model is not dependent on q_2 because there is no restriction on the low priority queue length. So, the number of busy servers will not increase when low priority customers enter the system. But this is not true when the low priority queue length is restricted to a finite number, i.e., when the M-cutoff point exists. We will discuss more about this in Section 3.3. Hence, from (3.11), and the above analysis, in D(N;N,∞,Q,Q) model we have

$$EW_\infty^\ell(N,q_1,q_2;k) = EW_\infty^\ell(N,q_1,k;k) = \frac{q_1+k}{N\mu-\lambda_1} \quad , \quad q_2 \geq k \geq 1 \ , \ q_1 \geq 0 \qquad (3.12)$$

Note that (3.12) have defined the conditional expected waiting times of low priority customers at all the feasible states.

3.2 D(N;R,∞;Q,Q) Model with R<N

For this model, (N-R) servers are reserved for high priority customer. That is, when the number of busy servers, b, is less than R, the arriving customer will be served immediately no matter what priority the customer is. But when b≥R, only the high priority customers are served whenever there is a server available. When all the servers are busy the high priority customers will be queued in the high priority queue. When a server completes service and returns to free it will first check the high priority queue. If the high priority queue is not empty then the first customer will start the service. If b<R (this will guarantee that the high priority queue is empty) and the low priority queue is not empty then the first low priority customer will start the service. In any other case, the server will remain free. Exhibit 3.2 is the transition diagram for D(N;R,∞;Q,Q) model.

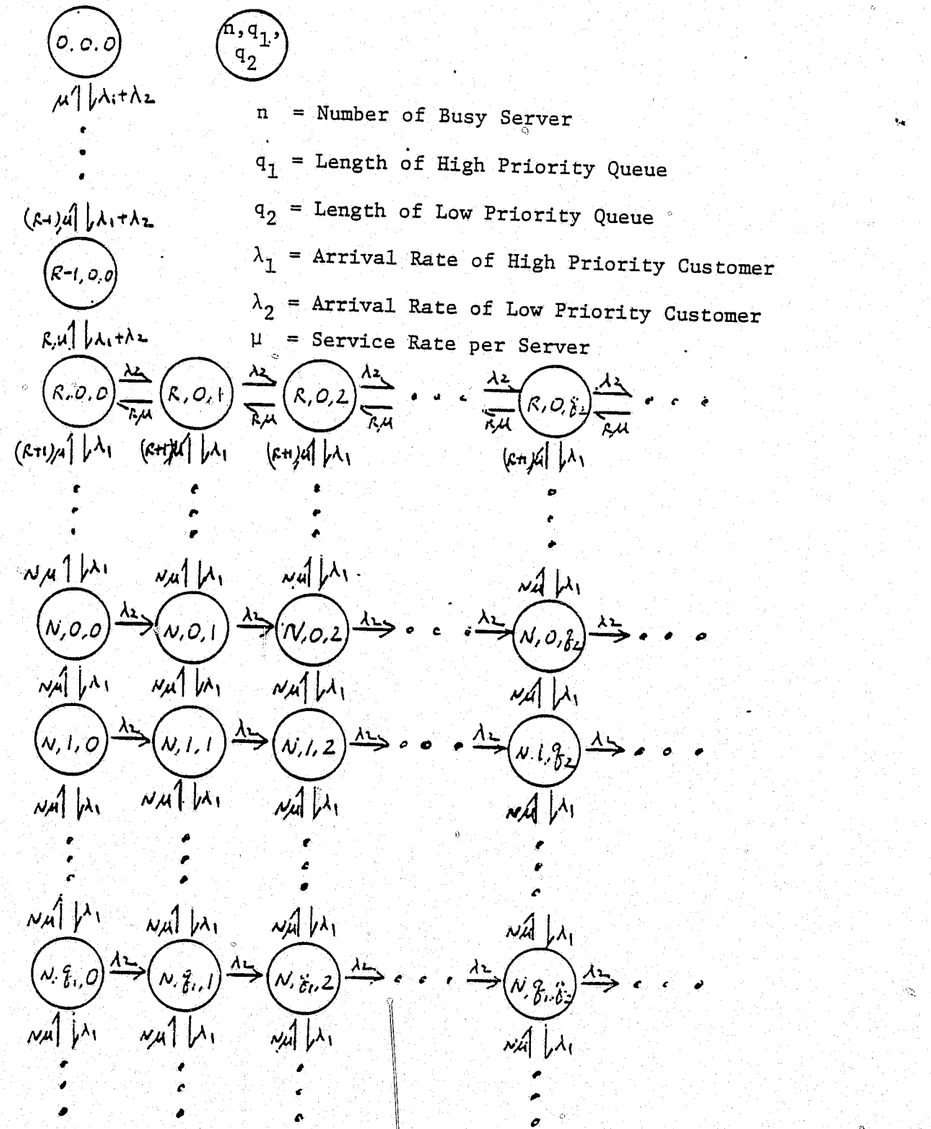


Exhibit 3.2 Transition Diagram for D(N;R,∞;Q,Q) Model

Steady State Probability Distribution

Taylor and Templeton [1980] worked on this model. They obtained the probability distribution of the number of busy servers. Hence, they obtained a formula for P(0,0,0)

$$P(0,0,0) = \left\{ \sum_{n=0}^{R-1} \rho^n/n! + (\rho^R/(R-1)!)\ S(R,N)/[R-\rho_2 S(R,N)] \right\}^{-1} \qquad (3.13)$$

where $S(R,N) = \rho_1^{-R} R! \left[\sum_{i=R}^{N-1} \rho_1^i/i! + (\rho_1^N/N!)\ N/(N-\rho_1) \right]$

We will use (3.13) to derive the steady state distribution for D(N;R,∞;Q,Q) model. We summarize the steps of finding these steady state probabilities below.

STEP 1) Compute P(n,0,0), n=1,...,R.

STEP 2) Use Theorem 3.1 to compute P(n,0,0), n=R+1,...,N and P(N,q_1,0), q_1=1,...,Q_1.

STEP 3) For each $q_2 \geq 1$, do

STEP 3.1) Compute P(R,0,q_2)

STEP 3.2) Compute P(n,0,q_2), n=R+1,...,N and P(N,q_1,q_2), q_1=1,...,Q_1

We outline these steps below.

STEP 1) Compute P(n,0,0), n=1,...,R.

$$P(n,0,0) = \frac{\rho^n}{n!} P(0,0,0) \quad , \quad n=1,\ldots,R \qquad (3.14)$$

STEP 2) Substitute q_1=1 in Theorem 3.1, we have

$$P(N,1,0) = x\ P(N,0,0) \qquad (3.15)$$

The balance equation at state (N,0,0) is

$$(N+\rho)P(N,0,0) = \rho_1 P(N-1,0,0) + NP(N,1,0)$$

Substitute (3.15) into above equation, we have

$$P(N-1,0,0) = P(N,0,0)\ [N(1-x)+\rho]/\rho_1$$
$$\equiv C_{N-1} P(N,0,0) \qquad (3.16)$$

where we have defined C_N=1 and $C_{N-1}=[N(1-x)+\rho]/\rho_1$

Doing the same analysis at state (N-1,0,0),...,(R+1,0,0), we got a sequence C_n defined recursively as below.

$$C_n = \frac{1}{\rho_1} [(n+1+\rho)\ C_{n+1} - (n+2)\ C_{n+2}] \quad , \quad n=N-2,\ldots,R \qquad (3.17)$$

Then we can express P(n,0,0), n=R,R+1,...,N in terms of P(N,0,0).

$$P(n,0,0) = C_n P(N,0,0) \quad , \quad n=R,\ldots,N \qquad (3.18)$$

For n=R, we have

$$P(R,0,0) = C_R P(N,0,0)$$

or, $$P(N,0,0) = P(R,0,0)/C_R \qquad (3.19)$$

Substitute (3.19) into (3.18) we will have P(n,0,0), n=R+1,...,N. Apply Theorem 3.1 we will have P(N,q_1,0), q_1=1,...,Q_1, where the Q_1 is determined in the same way as it is in D(N;N,∞;Q,Q) model.

STEP 3) For each $q_2 \geq 1$, do

STEP 3.1) Compute P(R,0,q_2)

$$P(R,0,q_2) = \left[\sum_{n=R}^{N} P(n,0,q_2-1) + \sum_{q_1=1}^{\infty} P(N,q_1,q_2-1) \right] (\rho_2/R)$$
$$\tilde{=} \left[\sum_{n=R}^{N} P(n,0,q_2-1) + \sum_{q_1=1}^{Q_1} P(N,q_1,q_2-1) \right] (\rho_2/R) \qquad (3.20)$$

STEP 3.2) Compute P(n,0,q_2), n=R+1,...,N and P(N,q_1,q_2), q_1=1,...,Q_1.

Ignoring the flow from state (N,Q_1+1,q_2) into state (N,Q_1,q_2) and the flow from state (N,Q_1,q_2) into state (N,Q_1+1,q_2) and forming the balance equation at state $(N,Q_1,q_2),\ldots,(N,0,q_2)$ successively, we obtained two recursive sequences A_{q_1} and B_{q_1}, $q_1=Q_1,\ldots,0$,

$$A_{q_1} = [(N+\rho)A_{q_1+1} - NA_{q_1+2}]/\rho_1$$

$$B_{q_1} = [(N+\rho)B_{q_1+1} - NB_{q_1+2} - \rho_2 P(N,q_1+1,q_2-1)]/\rho_1 \ , \ q_1=Q_1-2,Q_1-3,\ldots,0$$

with $A_{Q_1} = 1$, $B_{Q_1} = 0$, $A_{Q_1-1} = (N+\rho_2)/\rho_1$

and $B_{Q_1-1} = -\rho_2 P(N,Q_1,q_2-1)/\rho_1$.

Then we can express $P(N,q_1,q_2)$ in terms of $P(N,Q_1,q_2)$

$$P(N,q_1,q_2) = A_{q_1} P(N,Q_1,q_2) + B_{q_1} \quad , \quad q_1=0,1,\ldots,Q_1 \tag{3.21}$$

Let $C_N=A_0$ and $D_N=B_0$. Forming the balance equation at state $(n,0,q_2)$, $n=N-1,\ldots,R+1$ successively, we obtained two recursive sequence C_n and D_n, $n=N-1,\ldots,R$

$$C_n = [(n+1+\rho)C_{n+1} - (n+2)C_{n+2}]/\rho_1$$

$$D_n = [(n+1+\rho)D_{n+1} - (n+2)D_{n+2} - \rho_2 P(n+1,0,q_2-1)]/\rho_1 \quad , \quad n=N-2,\ldots,R$$

with $C_{N-1} = [(N+\rho)A_0 - NA_1]/\rho_1$ and $D_{N-1} = [(N+\rho)B_0 - NB_1 - \rho_2 P(N,0,q_2-1)]/\rho_1$

Then we can express $P(n,0,q_2)$ in terms of $P(N,Q_1,q_2)$

$$P(n,0,q_2) = C_n P(N,Q_1,q_2) + D_n \quad , \quad n=R,\ldots,N \tag{3.22}$$

For n=R in (3.22), we have

$$P(R,0,q_2) = C_R P(N,Q_1,q_2) + D_R$$

Or,
$$P(N,Q_1,q_2) = [P(R,0,q_2) - D_R]/C_R \tag{3.23}$$

Substitute (3.23) into (3.22) and (3.21) we will have $P(n,0,q_2)$, $n=R+1,\ldots,N$ and $P(N,q_1,q_2)$, $q_1=1,2,\ldots,Q_1$.

We can keep increasing q_2 by 1 and repeat STEP 3) until $P(R,0,q_2)$ is smaller than a prespecified value. Note that the steady state can be reached when $\rho_2<R$ and $\rho_1+\rho_2<N$.

Conditional Waiting Time Distribution for High Priority Customer

The conditional waiting time distribution for the k^{th} high priority customer is also Erlang distributed with parameters Nu and k. It is exactly the same as it is in the $D(N;N,\infty;Q,Q)$ model (see equation 3.8).

Conditional Expected Waiting Times for Low Priority Customer

The conditional waiting time distribution of low priority customer is not known at this moment. However, we found a very easy way to compute the expected value at a given state for each low priority customer.

Let $EW_\infty^\ell(n,q_1,q_2;k)$, $q_2\geq k$, denote the conditional expected waiting time of the k^{th} low priority customer at state (n,q_1,q_2). As we discussed earlier in Section 3.1, the conditional expected waiting time of the k^{th} low priority customer in $D(N;R,\infty;Q,Q)$ model is not dependent on q_2, the number of low priority customers in the queue. Hence we have the following expression in the $D(N;R,\infty;Q,Q)$ model.

$$EW_\infty^\ell(n,q_1,q_2;k) = EW_\infty^\ell(n,q_1,k;k), \ n=R,\ldots,N, \ q_1\geq 0 \text{ and } q_2\geq k\geq 1 \tag{3.24}$$

The system will not change state until one of the following events occurs.

(A) A server completes service and returns to free.

(B) A high priority customer arrives.

(C) A low priority customer arrives.

As soon as one of the three events occurs the system will move to a new state. We summarize the steps of finding the conditional expected waiting times of the first low priority customer below.

STEP 1) Express $EW_\infty^\ell(N,q_1,1;1)$, $q_1=0,1,2,\ldots$, in terms of $EW_\infty^\ell(N,0,1;1)$.

STEP 2) Express $EW_\infty^\ell(n,0,1;1)$, $n=N,N-1,\ldots,R+1$ in terms of $EW_\infty^\ell(n-1,0,1;1)$.

STEP 3) Compute $EW_\infty^\ell(n,0,1;1)$, $n=R,\ldots,N$ and $EW_\infty^\ell(N,q_1,1;1)$, $q_1=1,2\ldots$

We now outline these steps below.

STEP 1) Express $EW_\infty^\ell(N,q_1,1;1)$, $q_1=0,1,\ldots$, in terms of $EW_\infty^\ell(N,0,1;1)$.

Suppose the system is full and there is a low priority customer in the queue. Let T_1 be the time elapsed since the $(q_1+1)^{st}$ high priority customer arrives until the 1^{st} low priority customer goes into service. Then, $E(T_1)=EW_\infty^\ell(N,q_1+1,\cdot;1)$. Let T_2 be the time elapsed since the q_1^{th} high priority customer arrives until the 1^{st} low priority customer goes into service. Then $E(T_2)=EW_\infty^\ell(N,q_1,\cdot;1)$. Let $T=T_1-T_2$. Then we have

$$E(T) = E(T_1) - E(T_2) = EW_\infty^\ell(N,q_1+1,\cdot;1) - EW_\infty^\ell(N,q_1,\cdot;1)$$

By doing the busy period analysis on the high priority customer, [Davis, 1966], we have $E(T) = \frac{1}{N\mu-\lambda_1}$. That is,

$$EW_\infty^\ell(N,q_1+1,\cdot;1) - EW_\infty^\ell(N,q_1,\cdot;1) = \frac{1}{N\mu-\lambda_1}, \quad q_1\geq 0.$$

Hence

$$EW_\infty^\ell(N,q_1,\cdot;1) = EW_\infty^\ell(N,0,\cdot;1) + \frac{q_1}{N\mu-\lambda_1}, \quad q_1\geq 0 \tag{3.25}$$

STEP 2) Express $EW_\infty^\ell(n,0,1;1)$, $n=N,N-1,\ldots,R+1$, in terms of $EW_\infty^\ell(n-1,0,1;1)$.

Define $D_N = 1/(N\mu-\lambda_1)$ and

$$D_n = (1+\lambda_1 D_{n+1})/n\mu, \quad n=N-1,\ldots,R$$

then

$$EW_\infty^\ell(n,0,1;1) = EW_\infty^\ell(n-1,0,1;1) + D_n, \quad n=R,\ldots,N \tag{3.26}$$

where we have defined $EW_\infty^\ell(R-1,0,1;1)=0$.

STEP 3) Compute $EW_\infty^\ell(n,0,1;1)$, $n=R,\ldots,N$ and $EW_\infty^\ell(N,q_1,1;1)$, $q_1\geq 1$.

$$EW_\infty^\ell(R,0,1;1) = D_R \tag{3.27}$$

Substitute (3.27) into (3.26) and (3.25), we will have $EW_\infty^\ell(n,0,1;1)$, $n=R,\ldots,N$ and $EW_\infty^\ell(N,q_1,1;1)$, $q_1\geq 1$. We can extend this procedure to find the conditional expected waiting times of the k^{th} low priority customer, by defining the exact same sequence D_n, $n=R,\ldots,N$ in (3.26). Then

$$EW_\infty^\ell(R,0,k;k) = k\cdot D_R$$

$$EW_\infty^\ell(n,0,k;k) = EW_\infty^\ell(n-1,0,k;k) + D_n, \quad n=R+1,\ldots,N$$

$$EW_\infty^\ell(N,q_1,k;k) = EW_\infty^\ell(N,0,k;k) + \frac{k}{N\mu-\lambda_1}, \quad k=1,2,\ldots \tag{3.28}$$

Note that (3.28) together with (3.24) has defined all the conditional expected waiting times of each low priority customer at every state.

The unconditional expected waiting time of low priority in the $D(N;R,\infty;Q,Q)$ model is

$$EW_\infty^\ell = \sum_{n=R}^{N}\sum_{q_2=0}^{\infty} P(n,0,q_2)\, EW_\infty^\ell(n,0,q_2+1;q_2+1) + \sum_{q_1=1}^{\infty}\sum_{q_2=0}^{\infty} P(N,q_1,q_2)\, EW_\infty^\ell(N,q_1,q_2+1;q_2+1) \tag{3.29}$$

The reason for (3.29) is the same as it is for the D(N;R,∞;L,Q) model.

A program has been written to compute the steady state probabilities, the conditional expected waiting times for the low priority customers and the unconditional expected waiting time for low priority customer (using (3.29) with q_1 up to Q_1 and q_2 up to 50). Exhibit 3.3 is a list of comparisons between the theoretical values and the results obtained from the program.

N	R	λ_1	λ_2	μ	Q_1	Unconditional Expected Waiting Time of Low Priority Customer	
						Theoretical Value	Our Result
5	3	1	1	1	5	.3592	.3592
5	4	1	2	1	6	.4639	.4639
5	4	2	1	1	10	.4664	.4663
10	8	1	1	1	2	.0002	.0002
10	8	2	4	1	4	.1794	.1794
10	8	4	2	1	9	.1823	.1822
15	12	1	1	1	2	0	0
15	12	2	8	1	2	.2283	.2282
15	12	8	2	1	10	.2279	.2275
15	12	.05	.25	.034	2	2.3094	2.3094

Exhibit 3.3 Comparisons of the unconditional expected waiting time for low priority customer

As we can see from Exhibit 3.3, our result checks with the theoretical value. As expected, none of our results is greater than the theoretical value because we did not include all the terms in (3.29).

It is also interesting to compare the conditional expected waiting time of low priority customer in the D(N;R,∞;Q,Q) model to the conditional expected waiting time of low priority customer in the D(N;R,∞;L,Q) model. As expected, the former is

greater than the latter at all the corresponding states. For example, in the example at the bottom of page , the expected waiting time of the k^{th} low priority customer is .433k at state (3,k) in D(5;3,∞;L,Q) model. But in D(5;3,∞;Q,Q), by (3.28), the expected waiting time of the k^{th} low priority customer is .438k at state (3,0,k). The unconditional expected waiting time of low priority customer in D(N;R,∞;Q,Q) model is also greater than it is in the D(N;R,∞;L,Q) model. In the following exhibit, Exhibit 3.4, we list the comparisons of the unconditional expected waiting times of low priority customer between these two models.

N	R	λ_1	λ_2	μ	Unconditional Expected Waiting Time of Low Priority Customer	
					P(N;R,∞;L,Q)	D(N;R,∞;Q,Q)
5	2	1	1	1	.3418	.3592
5	4	1	2	1	.3649	.4639
5	4	2	1	1	.2246	.4664
10	8	1	1	1	.0002	.0002
10	8	2	4	1	.1679	.1794
10	8	4	2	1	.1263	.1823
15	12	1	1	1	0	0
15	12	2	8	1	.2272	.2283
15	12	8	2	1	.142	.2279
15	12	.05	.25	.034	2.3067	2.3094

Exhibit 3.4 Comparisons of the unconditional expected waiting time of low priority customer in D(N;R,∞;L,Q) model and D(N;R,∞;Q,Q) model

From Exhibit 3.4, we can see another property about the unconditional expected waiting time of low priority customer in the D(N;R,∞;L,Q) model. That is, for the same total arrival rate, the bigger the high priority customer

arrival rate, the less the expected waiting time of the low priority customer. The reason for this is that the actual input rate to D(N;R,∞;L,Q) model decreases with increase in high priority arrival rate and, at the mean time, the average of number of busy servers increases with increase in high priority arrival rate. But, in general, this is not true for the D(N;R,∞;Q,Q) model.

3.3 D(N;R,M;Q,Q) Model with R<N and M<∞

In this model, we reserved N-R servers for the high priority customer and, at the same time, we do not allow the low priority queue length exceeds M when the system is not full. The state space S is

$$S = \left\{ \begin{array}{lll} (n,q_1,q_2): & q_1=0, \ q_2=0 & \text{when } n<R \\ & q_1=0, \ q_2 \leq M & \text{when } R \leq n<N \\ & q_1 \geq 0, \ q_2 \geq 0 & \text{when } n=N \end{array} \right\}.$$

When a high priority customer arrives and finds the system is at state (n,q_1,q_2), he/she will be served immediately if n<N, otherwise he/she will be queued in the high priority queue. When a low priority customer arrives and finds the system is at state (n,q_1,q_2), then exactly one of the following four actions will be taken.

1) If n<R, then he/she will be served immediately.
2) If R≤n<N and q_2<M, then he/she will be queued in the low priority queue in the last position.
3) If R≤n<N and q_2=M, then he/she will be queued in the low priority queue in the last position and, at the same time, the first customer in the low priority queue will start the service.
4) If n=N, then he/she will be queued in the low priority queue in the last position.

When a server completes service and returns to free the system changes state from (n,q_1,q_2) to $(n-1,q_1,q_2)$, then exactly one of the following five actions will be taken:

1) If q_1>0, then the first high priority customer will start the service and the system will go into the state (N,q_1-1,q_2) instantaneously.
2) If q_1=0 and q_2>M, then the first low priority customer will start the service and the system will go into the state $(N,0,q_2-1)$ instantaneously.
3) If n-1≥R and q_1=0, q_2≤M, then the server remains free.
4) If n-1<R and q_2>0, then the first low priority customer will start the service and the system will go into the state $(n,0,q_2-1)$ instantaneously.
5) If n-1<R and q_2=0, then the server remains free.

Exhibit 3.5 is the transition diagram for the D(N;R,M;Q,Q) model.

Steady State Probability Distribution

In the following procedure we try to find all the $P(n,q_1;q_2)$ in terms of P(0,0,0) and then sum them up to 1 to get P(0,0,0). We summarize the steps below.

STEP 1) Find P(n,0,0), n=1,...,R in terms of P(0,0,0).

STEP 2) Use Theorem 3.1 to define a recursive sequence to find P(n,0,0), n=R+1,...,N in terms of P(0,0,0).

STEP 3) For a small prespecified number ε_1, to find Q_1 and use Theorem 3.1 to find $P(N,q_1,0)$, q_1=1,2,...,Q_1 in terms of P(0,0,0).

STEP 4) For 1≤q_2<M, do

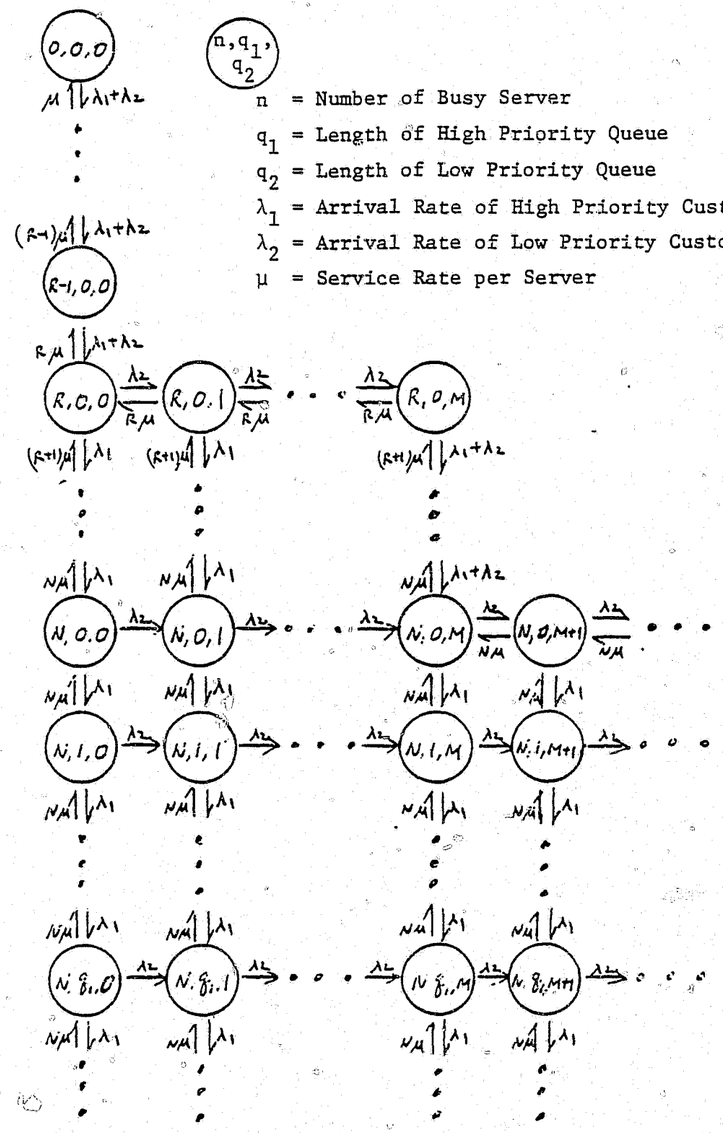


Exhibit 3.5 Transition Diagram for D(N;R,M;Q,Q) Model

STEP 4.1) Find $P(R,0,q_2)$ in terms of $P(0,0,0)$.

STEP 4.2) Define recursive sequences to find $P(n,0,q_2)$, $n=R+1,\ldots,N$ and $P(N,q_1,q_2)$, $q_1=1,\ldots,Q_1$ in terms of $P(0,0,0)$.

STEP 5) For $q_2=M$, do

STEP 5.1) Find $P(R,0,M)$ in terms of $P(0,0,0)$.

STEP 5.2) Define recursive sequences to find $P(n,0,M)$, $n=R+1,\ldots,N$ and $P(N,q_1,M)$, $q_1=1,2,\ldots,Q_1$ in terms of $P(0,0,0)$.

STEP 6) For $q_2>M$, do

STEP 6.1) Find $P(N,0,q_2)$ in terms of $P(0,0,0)$.

STEP 6.2) Define recursive sequences to find $P(N,q_1,q_2)$, $q_1=1,\ldots,Q_1$ in terms of $P(0,0,0)$.

STEP 6.3) STOPPING criterion for STEP 6). We keep increasing q_2 by 1 until $P(N,0,q_2)<\varepsilon_2 P(0,0,0)$, where ε_2 is a small number.

STEP 7) Express the probabilities for the rest of state (i.e., $q_1>Q_1$ and $q_2>Q_2$ in terms of $P(0,0,0)$).

STEP 8) Sum all of these probabilities to 1 to obtain $P(0,0,0)$.

We outline this procedure below.

STEP 1) Find $P(n,0,0)$, $n=1,\ldots,R$.

$$P(n,0,0) = \frac{\rho^n}{n!} P(0,0,0) \quad , \quad n=1,2,\ldots,R \tag{3.30}$$

STEP 2) Form the balance equation at state $(N,0,0),(N-1,0,0),\ldots,(R+1,0,0)$ and apply Theorem 3.1, we obtained a recursive sequence C_n

$$C_n = \frac{1}{\rho_1}[(n+1+\rho)C_{n+1} - (n+2)C_{n+2}] \quad , \quad n=N-2,N-3,\ldots,R$$

with $C_N=1$ and $C_{N-1} = [N(1-x)+\rho]/\rho_1$ where x is defined in Theorem 3.1.

Then we have

$$P(n,0,0) = C_n P(N,0,0) \quad , \quad n=R,\ldots,N \tag{3.31}$$

For n=R in (3.31), we have

$$P(R,0,0) = C_R P(N,0,0)$$

or $$P(N,0,0) = P(R,0,0)/C_R \tag{3.32}$$

But P(R,0,0) is already found in terms of P(0,0,0) in (3.30), hence we can substitute (3.32) into (3.31) and obtain P(n,0,0), n=R+1,...,N in terms of P(0,0,0).

STEP 3) As we did before, we can find the integer Q_1 such that $P(N,Q_1+1,0) < \varepsilon_1 P(0,0,0)$ and $P(N,Q_1,0) \geq \varepsilon_1 P(0,0,0)$ for a prespecified small number ε_1, for example 10^{-7}. Then we can apply Theorem 3.1 to find $D(N,q_1,0)$, $q=1,\ldots,Q_1$ in terms of P(0,0,0).

STEP 4) For $1 \leq q_2 < M$, do

STEP 4.1) $$P(R,0,q_2) = \left[\sum_{n=R}^{N} P(n,0,q_2-1) + \sum_{q_1=1}^{\infty} P(N,q_1,q_2-1) \right] (\rho_2/R)$$

$$\tilde{=} \left[\sum_{n=R}^{N} P(n,0,q_2-1) + \sum_{q_1=1}^{Q_1} P(N,q_1,q_2-1) \right] (\rho_2/R) \tag{3.33}$$

STEP 4.2) By ignoring the flow from state (N,Q_1+1,q_2) to state (N,Q_1,q_2) and the flow from state (N,Q_1,q_2) to state (N,Q_1+1,q_2). We can obtain four recursive sequences C_{q_1}, D_{q_1}, A_n and B_n, $q_1=Q_1,\ldots,1$, $n=N,\ldots,R$ from the balance equation at state (N,q_1,q_2), $q_1=Q_1,\ldots,1$ and $(n,0,q_2)$, $n=N,\ldots,R$. The sequences are defined below.

$$C_{Q_1} = 1$$

$$C_{Q_1-1} = (N+\rho_2)/\rho_1$$

$$C_{q_1} = [(N+\rho)C_{q_1+1} - NC_{q_1+2}]/\rho_1 \quad , \quad q_1=Q_1-2,\ldots,1$$

$$D_{Q_1} = 0$$

$$D_{Q_1-1} = -\rho_2 P(N,Q_1,q_2-1)/\rho_1$$

$$D_{q_1} = [(N+\rho)D_{q_1+1} - ND_{q_1+2} - \rho_2 P(N,q_1+1,q_2-1)]/\rho_1 \quad , \quad q_1=Q_1-2,\ldots,1$$

$$A_N = [(N+\rho)C_1 - NC_2]/\rho_1$$

$$A_{N-1} = [(N+\rho)A_N - NC_1]/\rho_1$$

$$A_n = [(n+1+\rho)A_{n+1} - (n+2)A_{n+2}]/\rho_1 \quad , \quad n=N-2,\ldots,R$$

$$B_N = [(N+\rho)D_1 - ND_2 - \rho_2 P(N,1,q_2-1)]/\rho_1$$

$$B_{N-1} = [(N+\rho)B_N - ND_1 - \rho_2 P(N,0,q_2-1)]/\rho_1$$

and

$$B_n = [(n+1+\rho)B_{n+1} - (n+2)B_{n+2} - \rho_2 P(n+1,0,q_2-1)]/\rho_1 \quad , \quad n=N-2,\ldots,R$$

Then we have

$$P(n,0,q_2) = A_n P(N,Q_1,q_2) + B_n \quad , \quad n=R,\ldots,N$$

and

$$P(N,q_1,q_2) = C_{q_1} P(N,Q_1,q_2) + D_{q_1} \quad , \quad q_1=1,\ldots,Q_1 \tag{3.34}$$

For n=R in (3.34) we have

$$P(R,0,q_2) = A_R P(N,Q_1,q_2) + B_R$$

or

$$P(N,Q_1,q_2) = [P(R,0,q_2) - B_R]/A_R \tag{3.35}$$

Since we already know $P(R,0,q_2)$ in (3.33), we can substitute (3.35) into (3.34) to get the probabilities in terms of $P(0,0,0)$.

STEP 5) $q_2=M$, do

STEP 5.1)

$$P(R,0,M) = \frac{\rho_2}{R}\left[\sum_{n=R}^{N} P(n,0,M-1) + \sum_{q_1=1}^{\infty} P(N,q_1,M-1)\right]$$

$$\simeq \frac{\rho_2}{R}\left[\sum_{n=R}^{N} P(n,0,M-1) + \sum_{q_1=1}^{Q_1} P(N,q_1,M-1)\right] \qquad (3.36)$$

STEP 5.2) By ignoring the flow from state (N,Q_1+1,M) into the state (N,Q_1,M) and the flow from state (N,Q_1,M) into the state (N,Q_1+1,M), we can generate four recursive sequences, C_{q_1}, D_{q_1}, A_n and B_n, $q_1=Q_1,\ldots,1$, $n=N,\ldots,R$ from the balance equation at state (N,q_1,M), $q_1=Q_1,\ldots,1$ and state $(n,0,M)$, $n=N,\ldots,R$. The sequences are defined as below.

$$C_{Q_1} = 1$$

$$C_{Q_1-1} = (N+\rho_2)/\rho_1$$

$$C_{q_1} = [(N+\rho)C_{q_1+1} - NC_{q_1+2}]/\rho_1 \quad , \quad q_1=Q_1-2,\ldots,1$$

$$D_{Q_1} = 0$$

$$D_{Q_1-1} = -\rho_2 P(N,Q_1,M-1)/\rho_1$$

$$D_{q_1} = [(N+\rho)D_{q_1+1} - ND_{q_1+2} - \rho_2 P(N,q_1+1,M-1)]/\rho_1 \quad , \quad q_1=Q_1-2,\ldots,1$$

$$A_N = [(N+\rho)C_1 - NC_2]/\rho_1$$

$$A_{N-1} = [(N+\rho)A_N - NC_1 - \rho_2(A_N + \sum_{q_1=1}^{Q_1} C_{q_1})]/\rho$$

$$A_n = [(n+1+\rho)A_{n+1} - (n+2)A_{n+2}]/\rho \quad , \quad n=N-2,\ldots,R$$

$$B_N = [(N+\rho)D_1 - ND_2 - \rho_2 P(N,1,M-1)/\rho_1$$

$$B_{N-1} = [(N+\rho)B_N - ND_1 - \rho_2 P(N,0,M-1) - \rho_2(B_N + \sum_{q_1=1}^{Q_1} D_{q_1})]/\rho$$

$$B_n = [(n+1+\rho)B_{n+1} - (n+2)B_{n+2} - \rho_2 P(n+1,0,M-1)]/\rho \quad , \quad n=N-2,\ldots,R$$

Then we have

$$P(n,0,M) = A_n P(N,Q_1,M) + B_n \quad , \quad n=R,\ldots,N$$

and $$P(N,q_1,M) = C_{q_1} P(N,Q_1,M) + D_{q_1} \quad , \quad q_1=1,\ldots,Q_1 \qquad (3.37)$$

For, n=R in (3.37) we have

$$P(R,0,M) = A_R P(N,Q_1,M) + B_R$$

or $$P(N,Q_1,M) = [P(R,0,M) - B_R]/A_R \qquad (3.38)$$

Since $P(R,0,M)$ is already known from Step 5.1), we can substitute (3.38) into (3.37) to get the probabilities in terms of $P(0,0,0)$.

STEP 6) For $q_2>M$, do

STEP 6.1) $$P(N,0,q_2) = (\rho_2/N) \sum_{q_1=0}^{\infty} P(N,q_1,q_2-1)$$

$$= (\rho_2/N) \sum_{q_1=0}^{Q_1} P(N,q_1,q_2-1) \qquad (3.39)$$

STEP 6.2) By doing the same analysis in the Step 5.2), we have two recursive sequences, C_{q_1} and D_{q_1}, $q_1=Q_1,\ldots,0$. The sequences C_{q_1} and D_{q_1} is defined recursively below.

$$C_{Q_1} = 1$$

$$C_{Q_1-1} = (N+\rho_2)/\rho_1$$

$$C_{q_1} = [(N+\rho)C_{q_1+1} - NC_{q_1+2}]/\rho_1 \quad , \quad q_1=Q_1-2,\ldots,0$$

$$D_{Q_1} = 0$$

$$D_{Q_1} = -\rho_2 P(N,Q_1,q_2-1)/\rho_1$$

and $D_{q_1} = [(N+\rho)D_{q_1+1} - N D_{q_1+2} - \rho_2 P(N,q_1+1,q_2-1)]/\rho_1$, $q_1=Q_1-2,\ldots,0$

Then we have

$$P(N,q_1,q_2) = C_{q_1} P(N,Q_1,q_2) + D_{q_1} \quad , \quad q_1=0,.,\ldots,Q_1 \qquad (3.40)$$

For $q_1=0$ in (3.42), we have

$$P(N,0,q_2) = C_0 P(N,Q_1,q_2) + D_0$$

or $$P(N,Q_1,q_2) = [P(N,0,q_2) - D_0]/C_0 \qquad (3.41)$$

Since we already found $P(N,0,q_2)$ in (3.39), we can substitute (3.41) into (3.40) to get $P(N,q_1,q_2)$, $q_1=1,\ldots,Q_1$.

STEP 6.3) We keep increasing q_2 by 1 each time until $P(N,0,q_2)<\varepsilon_2 P(0,0,0)$ where ε_2 is a small number, for example, 10^{-4}. Let the final value of q_2 be Q_2.

STEP 7) First we consider the probabilities at the states (N,q_1,q_2), $q_1=0,1,\ldots,Q_1$ and $q_2>Q_2$. Empirically it shows that, for every $q_1=0,1,\ldots,Q_1$, the ratio of $P(N,q_1,q_2)/P(N,q_1,q_2-1)$ converges to a constant as $q_2\to\infty$. We cannot proof this at this moment. We use it here without the proof. Let $R(q_1)=P(N,q_1,Q_2)/P(N,q_1,Q_2-1)$ for $q_1=0,1,\ldots,Q_1$. Then,

$$\sum_{q_2=Q_2+1}^{\infty} P(N,q_1,q_2) \tilde{=} P(N,q_1,Q_2) * R(q_1)/(1-R(q_1)) \quad , \quad q_1=0,1,\ldots,Q_1 \qquad (3.42)$$

Second, consider the probabilities at the states (N,q_1,q_2), $q_1>Q_1$, $q_2=0,1,\ldots,Q_2$. Empirically it also shows that for every $q_2=0,1,\ldots,Q_2$, the ratio of $P(N,q_1,q_2)/P(N,q_1-1,q_2)$ converges to a constant as $q_1\to\infty$. We also use it here without the proof. Let $S(q_2)=P(N,Q_1,q_2)/P(N,Q_1-1,q_2)$, $q_2=0,1,\ldots,Q_2$. Then

$$\sum_{q_1=Q_1+1}^{\infty} P(N,q_1,q_2) \tilde{=} P(N,Q_1,q_2)S(q_2)/(1-S(q_2)), \quad q_2=0,1,\ldots,Q_2 \qquad (3.43)$$

Third, we consider the probabilities at the states $P(N,q_1,q_2)$, $q_1>Q_1$, $q_2>Q_2$.

$$\sum_{q_1=Q_1+1}^{\infty} \sum_{q_2=Q_2+1}^{\infty} P(N,q_1,q_2) \tilde{=} P(N,Q_1,Q_2)R(Q_1)S(Q_2)/[(1-R(Q_1))(1-S(Q_2))]$$

STEP 8). $\sum_{n,q_1,q_2} P(n,q_1,q_2) = sP(0,0,0) = 1$

$$P(0,0,0) = 1/s$$

Note that we can omit STEP 7) if the ε_2 in STEP 6.3) is small enough, say $\varepsilon_2<10^{-7}$. The reason we have STEP 7) here is that we will have less iterations on q_2 if we choose a bigger ε_2. This algorithm has been programmed in FORTRAN in double precision with $\varepsilon_1=10^{-7}$ in STEP 3) and $\varepsilon_2=10^{-4}$ in STEP 6.3). The $P(0,0,0)$ we get from the program is at least correct up to the 6^{th} significant digit for any model. Exhibit 3.6 is the block diagram of finding the $P(0,0,0)$ in the $D(N;R,M;Q,Q)$ model.

As we can see from the block diagram, the number of operations is of the order of $O(Q_2(N-R+1+Q_1))$. In most of the models, Q_1 is less than 10 and Q_2 is less than 50.

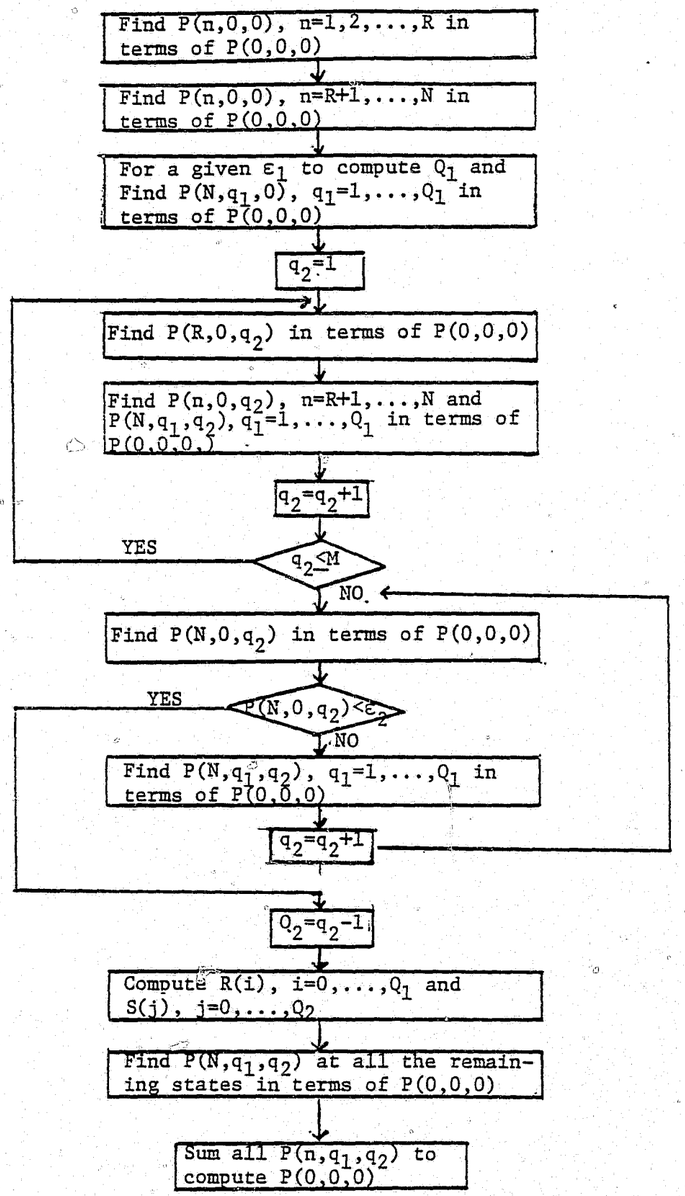


Exhibit 3.6 Block diagram of finding P(0,0,0) in the D(N;R,M;Q,Q) model

The P(0,0,0) in the D(N;R,M;Q,Q) model should decrease as M increase (because it has more states when M increases). For the special case R=N-1, the P(0,0,0) in the D(N;N-1,M;Q,Q) model is bounded below by the P(0,0,0) in the D(N;N-1;∞;Q,Q) model and is bounded above by the P(0,0,0) in the D(N;N-1,0;Q,Q) model which is equivalent to D(N;N,∞;Q,Q) model. And, when M→∞, $P_M(0,0,0) \to P_\infty(0,0,0)$, where $P_M(0,0,0)$ is the steady state probability at state (0,0,0) in the D(N;N-1,M;Q,Q) model. Since we know the exact values of the bounds, $P_0(0,0,0)$ and $P_\infty(0,0,0)$, we would like to compare the $P_M(0,0,0)$, M=1,2,... to the bounds $P_0(0,0,0)$ and $P_\infty(0,0,0)$. Exhibit 3.7 is a list of the comparisons of P(0,0,0)'s with M=0,1,5,10,20 and ∞ in D(N;N-1,M;Q,Q) model. As we can see from Exhibit 3.7, the $P_M(0,0,0)$, M=1,2,... is between $P_0(0,0,0)$ and $P_\infty(0,0,0)$ and it decreases to $P_\infty(0,0,0)$ as M increases to ∞.

N	R	λ_1	λ_2	μ	M	$P_M(0,0,0)$
5	4	1	2	1	0	.46647E-1
5	4	1	2	1	1	.44615E-1
5	4	1	2	1	5	.41893E-1
5	4	1	2	1	10	.41334E-1
5	4	1	2	1	20	.41242E-1
5	4	1	2	1	∞	.41237E-1
10	9	2	6	1	0	.27657E-3
10	9	2	6	1	1	.26628E-3
10	9	2	6	1	5	.24322E-3
10	9	2	6	1	10	.23070E-3
10	9	2	6	1	20	.22261E-3
10	9	2	6	1	∞	.22025E-3

Exhibit 3.7 Comparisons of P(0,0,0)'s for different M-cutoff point in D(N;N-1,M;Q,Q) model

Waiting Time Distribution for High Priority Customer

The waiting time density function for the k^{th} high priority customer in the queue is also Erlang distributed with parameters $N\mu$ and k. It is exactly the same as in the D(N;N,∞;Q,Q) model (see equation 3.8).

Conditional Expected Waiting Time for Low Priority Customer

Let $EW_M^{\ell}(n,q_1,q_2;k)$, $q_2 \geq k$, denote the conditional expected waiting time of the k^{th} low priority customer at state (n,q_1,q_2) in the D(N;R,M;Q,Q) model. In this section, we will derive a recursive formula to compute $EW_M^{\ell}(n,q_1,q_2;k)$ for each $k \leq q_2$ on all the feasible states (n,q_1,q_2).

Following that we will carry out an example to show how it works. Note that, for R=N-1, the unconditional expected waiting time of low priority customer in D(N;N-1,M;Q,Q) model should be bounded above and below by the unconditional expected waiting times of low priority customer in D(N;N-1;∞;Q,Q) model and D(N;N,∞;Q,Q) model respectively for M=1,2,... And, for R=N-1, $EW_M^{\ell}(n,q_1,q_2;k)$ in D(N;N-1,M;Q,Q) model will approach to $EW_{\infty}^{\ell}(n,q_1,q_2;k)$ in D(N;N-1,∞;Q,Q) model as M→∞. Since we have the exact value of the bounds, we want to compare our results to the bounds. Following that we will discuss an upper bound and a lower bound on the conditional expected waiting times. Finally, we examine an example, D(15;12,5;Q,Q), with the same rates as they are in Section 2.3.

Now, we start to compute $EW_M^{\ell}(n,q_1,q_2;k)$, $k \leq q_2$. Since the low priority queue length is restricted to M when the system is not full, $EW_M^{\ell}(n,q_1,q_2;k)$ will depend on q_2 as well as on n, q_1 and k. It is very easy to see this property at state (N,0,M) and state (N,0,M+1). At state (N,0,M+1), the first low priority customer will start the service if no high priority customer enters the system before a server completes service. But at state (N,0,M), the first low priority customer still wait in the queue when a server completes service. Hence,

equation (3.25) will not hold in general. But it still holds when $q_2 \geq M+k$, k=1,2,..., because when $q_2 \geq M+k$ the D(N;R,M;Q,Q) model works just like the D(N;N,∞;Q,Q) model. As it is in the D(N;R,M;L,Q) model, we have to find a state such that the expected waiting time at that state is known to start the recursive process for each k.

For k=1, we start at state (N,0,M+1). At state (N,0,M+1), there are M+1 customer waiting in the low priority queue, no customer waiting in the high priority queue and all the N servers are busy. Let T be the time elapsed since the system went into state (N,0,M+1) until the first low priority customer goes into service. Then, $E(T)=EW_M^{\ell}(N,0,M+1;1)$. Under D(N;R,M;Q,Q) model, the density function of T can be expressed in the transformed space

$$f_T(s) = (s+N\mu+\lambda_1 - \sqrt{(s+N\mu+\lambda_1)^2-4N\mu\lambda_1})/2\lambda_1 .$$

And, $E(T)=1/(N\mu-\lambda_1)$. Hence, we have

$$EW_M^{\ell}(N,0,M+1;1) = E(T) = 1/(N\mu-\lambda_1) \qquad (3.44)$$

At state $(N,q_1,M+1)$, there are (M+1) customer waiting in the low priority queue, q_1 customer waiting in the high priority queue and all the N servers are busy. Let T_{q_1} be the time elapsed since the system went into state $(N,q_1,M+1)$ until the first low priority customer goes into service. The distribution of T_{q_1} is the $(q_1+1)^{st}$ fold convolution of the distribution of T and the expected value, $E(T_{q_1})$, is equal to $(q_1+1)/(N\mu-\lambda_1)$. Hence

$$EW_M^{\ell}(N,q_1,M+1;1) = E(T_{q_1}) = (q_1+1)/(N\mu-\lambda_1) \quad , \quad q_1=1,2,... \qquad (3.45)$$

In applying (3.12) for $q_2 \geq M+1$, we have

CONTINUED

1 OF 2

$$EW_M^{\ell}(N,q_1,q_2;1) = (q_1+1)/(N\mu-\lambda_1), \quad q_1 \geq 0, \; q_2 \geq M+1 \tag{3.46}$$

Now, we have found the conditional expected waiting times of the 1st low priority customer for $q_2 > M$. We want to proceed this for $q_2 = M$.

At state (R,0,M), we have

$$EW_M^{\ell}(R,0,M;1) = \frac{1}{R\mu+\lambda_1+\lambda_2} + \frac{\lambda_1}{R\mu+\lambda_1+\lambda_2} EW_M^{\ell}(R+1,0,M;1)$$

$$\equiv A_R + B_R \, EW_M^{\ell}(R+1,0,M;1) \tag{3.47}$$

where we have defined $A_R = \frac{1}{\mu(R+\rho)}$ and $B_R = \frac{\rho_1}{R+\rho}$.

Follow the same analysis at state (n,0,M), n=R+1,...,N-1, we have

$$EW_M^{\ell}(n,0,M;1) = A_n + B_n EW_M^{\ell}(n+1,0,M;1), \; n=R+1,\ldots,N-1 \tag{3.48}$$

where $A_n = \frac{1/\mu+nA_{n-1}}{n(1-B_{n-1})+\rho}$ and $B_n = \frac{\rho_1}{n(1-B_{n-1})+\rho}$

At state (N,0,M),

$$EW_M^{\ell}(N,0,M;1) = \frac{1}{N\mu+\lambda_1+\lambda_2} + \frac{N\mu}{N\mu+\lambda_1+\lambda_2} EW_M^{\ell}(N-1,0,M;1)$$

$$+ \frac{\lambda_1}{N\mu+\lambda_1+\lambda_2} EW_M^{\ell}(N,1,M;1) + \frac{\lambda_2}{N\mu+\lambda_1+\lambda_2} EW_M^{\ell}(N,0,M+1;1)$$

Substitute the value $EW_M^{\ell}(N,0,M+1;1)$ in (3.46) into the above equation, we have

$$EW_M^{\ell}(N,0,M;1) = A_N + B_N \, EW_M^{\ell}(N,1,M;1) \tag{3.49}$$

where $A_N = \frac{1/\mu+NA_{N-1}+\rho_2/(N\mu-\lambda_1)}{N(1-B_{N-1})+\rho}$ and $B_N = \frac{\rho_1}{N(1-B_{N-1})+\rho}$

Do the same analysis at state (N,q_1,M), $q_1=1,\ldots,Q_1$, we have

$$EW_M^{\ell}(N,q_1,M;1) = C_{q_1} + D_{q_1} EW_M^{\ell}(N,q_1+1,M;1), \; q_1=1,2,\ldots \tag{3.50}$$

with $C_1 = \frac{1/\mu+NA_N+2\rho_2/(N\mu-\lambda_1)}{N(1-B_N)+\rho}$, $D_1 = \frac{\rho_1}{N(1-B_N)+\rho}$

$$C_{q_1} = \frac{1/\mu+NC_{q_1-1}+\rho_2(q_1+1)/(N\mu-\lambda_1)}{N(1-D_{q_1-1})+\rho} \text{ and } D_{q_1} = \frac{1}{N(1-D_{q_1-1})+\rho}, \; q_1=2,3,\ldots$$

As we can see from equation (3.50), we need a value of $EW_M^{\ell}(N,q_1+1,M;1)$ to compute $EW_M^{\ell}(N,q_1,M;1)$. We found that $EW_M^{\ell}(N,q_1+1,M;1) - EW_M^{\ell}(N,q_1,M;1) \to 1/(N\mu-\lambda_1)$ as $q_1 \to \infty$. We prove this in the following lemma.

Lemma 3.1: The sequence D_{q_1} in (3.50) approach to

$$[N+\rho-\sqrt{(N+\rho)^2-4N\rho_1}\,]/2N \quad \text{as } q_1 \to \infty$$

Proof: In (3.50) we have, $0 < D_{q_1} < 1$ for every q_1 and $D_{q_1} = \frac{\rho_1}{N(1-D_{q_1-1})+\rho}$ is monotone sequence of q_1.

Let $D_{q_1} \to D$ as $q_1 \to \infty$, then

$$ND^2 - (N+\rho)D + \rho_1 = 0$$

or $$D = [(N+\rho) - \sqrt{(N+\rho)^2 - 4N\rho_1}\,]/2N \qquad \text{Q.E.D.}$$

Lemma 3.2: The sequence $C_{q_1+1} - C_{q_1} \to \frac{\rho_2}{(\rho-ND)(N\mu-\lambda_1)}$ as $q_1 \to \infty$ where D is defined in Lemma 3.1.

Proof: From (3.50), we have

$$C_{q_1+1} - C_{q_1} = \frac{1/\mu+NC_{q_1}+\rho_2(q_1+2)/(N\mu-\lambda_1)}{N(1-D_{q_1})+\rho} - \frac{1/\mu+NC_{q_1-1}+\rho_2(q_1+1)/(N\mu-\lambda_1)}{N(1-D_{q_1-1})+\rho}$$

From Lemma 3.1, $D_{q_1} \to D$ as $q_1 \to \infty$.

Let $C_{q_1+1} - C_{q_1} \to C$ when $q_1 \to \infty$, then

$$C = \frac{NC+\rho_2/(N\mu-\lambda_1)}{N(1-D)+\rho}$$

or $$C = \rho_2/[(\rho-ND)(N\mu-\lambda_1)] \qquad \text{Q.E.D.}$$

Lemma 3.3. $EW_M^{\ell}(N,q_1+1,M;1) - EW_M^{\ell}(N,q_1,M;1) \to \frac{1}{N\mu-\lambda_1}$ as $q_1 \to \infty$.

Proof: From (3.50), we have

$$EW_M^{\ell}(N,q_1+1,M;1) - EW_M^{\ell}(N,q_1,M;1)$$
$$= (C_{q_1+1} - C_{q_1}) + D_{q_1+1}\, EW_M^{\ell}(N,q_1,M;1) - D_{q_1}\, EW_M^{\ell}(N,q_1-1,M;1)$$

Let $EW_M^{\ell}(N,q_1+1,M;1) - EW_M^{\ell}(N,q_1,M;1) \to E$ as $q_1 \to \infty$.

Then

$$E = C + DE$$

or $$E = \frac{C}{1-D}$$

Substitute C and D in Lemma 3.1 and 3.2, respectively, we have

$$E = \frac{1}{N\mu-\lambda_1} \qquad \text{Q.E.D.}$$

If we pick a value Q_1 and apply Lemma 3.3, we have

$$EW_M^{\ell}(N,Q_1,M;1) \cong EW_M^{\ell}(N,Q_1+1,M;1) - \frac{1}{N\mu-\lambda_1} \qquad (3.51)$$

From equation (3.46), we have

$$EW_M^{\ell}(N,Q_1,M;1) = C_{Q_1} + D_{Q_1}\, EW_M^{\ell}(N,Q_1+1,M;1) \qquad (3.52)$$

Equate (3.51) to (3.52), we have

$$EW_M^{\ell}(N,Q_1+1,M) = (C_{Q_1} + \frac{1}{N\mu-\lambda_1})/(1-D_{Q_1}) \qquad (3.53)$$

Substitute (3.53) backwards into (3.50), (3.49), (3.48) and (3.47) we will have the conditional expected waiting times of the first low priority at those states with low priority queue length equals M.

In general, for each q_2=M-1,M-2,...,1, we have

$$EW_M^{\ell}(n,0,q_2;1) = A_n + B_n EW_M^{\ell}(n+1,0,q_2;1) \ , \ n=R,\ldots,N-1$$

$$EW_M^{\ell}(N,0,q_2;1) = A_N + B_N EW_M^{\ell}(N,1,q_2;1)$$

$$EW_M^{\ell}(N,q_1,q_2;1) = C_{q_1} + D_{q_1} EW_M^{\ell}(N,q_1+1,q_2;1) \ , \ q_1=1,2,\ldots,Q_1 \qquad (3.54)$$

where

$$A_R = [1/\mu + \rho_2 EW_M^{\ell}(R,0,q_2+1;1)]/(R+\rho)$$

$$B_R = \rho_1/(R+\rho)$$

$$A_n = [1/\mu+nA_{n-1}+\rho_2 EW_M^{\ell}(n,0,q_2+1;1)]/[n(1-B_{n-1}) + \rho] \ , \ n=R+1,\ldots,N$$

$$B_n = \rho_1/[n(1-B_{n-1}) + \rho] \ , \ n=R+1,\ldots,N$$

$$C_1 = [1/\mu+NA_N+\rho_2 EW_M^{\ell}(N,1,q_2+1;1)]/[N(1-B_N)+\rho]$$

$$D_1 = \rho_1/[N(1-B_N) + \rho]$$

$$C_{q_1} = [1/\mu+NC_{q_1-1}+\rho_2 EW_M^{\ell}(N,q_1,q_2+1;1)]/[N(1-D_{q_1-1})+\rho] \ , \ q_1=2,\ldots,Q_1$$

and

$$D_{q_1} = \rho_1/[N(1-D_{q_1-1}) + \rho] \ , \ q_1=2,\ldots,Q_1$$

As we can see from (3.54), again, we need a value of $EW_M^{\ell}(N,q_1+1,q_2;1)$ to compute $EW_M^{\ell}(N,q_1,q_2;1)$. Lemma 3.3 still holds for $q_2<M$. The sequence D_{q_1} is exactly

the same as it is at q_2=M. Hence, Lemma 3.1 holds for every q_2=M-1,...,1. We now prove Lemma 3.2 holds for q_2=M-1.

Lemma 3.4. When q_2=M-1, $\lim_{q_1\to\infty} [C_{q_1+1}-C_{q_1}] = \rho_2/[(\rho-ND)(N\mu-\lambda_1)]$, where C_{q_1} is defined in (3.54) and D is defined in Lemma 3.1.

Proof: From (3.54) we have

$$C_{q_1+1} - C_{q_1} = \frac{1/\mu+NC_{q_1}+\rho_2 EW_M^{\ell}(N,q_1+1,M;1)}{N(1-D_q)+\rho} - \frac{1/\mu+NC_{q_1-1}+\rho_2 EW_M^{\ell}(N,q_1,M;1)}{N(1-D_{q-1})+\rho}$$

In applying Lemma 3.3 and let $q_1\to\infty$, we have

$$\lim_{q_1\to\infty} [C_{q_1+1}-C_{q_1}] = C = \frac{NC+\rho_2/(N\mu-\lambda_1)}{N(1-D)+\rho}$$

or $C = \rho_2/[(\rho-ND)(N\mu-\lambda_1)]$ Q.E.D.

Use Lemma 3.4 and the techniques we used in the proof of Lemma 3.3, we have

$$\lim_{q_1\to\infty} [EW_M^{\ell}(N,q_1+1,M-1;1) - EW_M^{\ell}(N,q_1,M-1;1)] = 1/(N\mu-\lambda_1)$$

We can repeat this process until q_2=1. We state this in the following Lemma.

Lemma 3.5. $\lim_{q_1\to\infty} [EW_M^{\ell}(N,q_1+1,q_2;1) - EW_M^{\ell}(N,q_1,q_2;1)] = 1/(N\mu-\lambda_1)$

for $1\le q_2\le M$.

Hence, we can pick a number Q_1 such that

$$EW_M^{\ell}(N,Q_1,q_2;1) = C_{Q_1} + D_{Q_1} EW_M^{\ell}(N,Q_1+1,q_2;1) \quad \text{(by 3.54)}$$

$$\cong EW_M^{\ell}(N,Q_1+1,q_2;1) - \frac{1}{N\mu-\lambda_1} \quad \text{(by Lemma 3.5)}$$

or $EW_M^{\ell}(N,Q_1+1,q_2;1) = [C_{Q_1} + 1/(N\mu-\lambda_1)]/(1 - D_{Q_1})$ (3.55)

Substitute (3.55) into (3.54), we will have the conditional expected waiting times of the 1st low priority customer.

Exhibit 3.8 is the block diagram of finding the conditional expected waiting times of the 1st low priority customer. As we can see from the block diagram, the number of operation to compute these expected waiting times is of the order of $O(M(Q_1+N-R+1))$. We can generalize this procedure to the k^{th} low priority customer, $k\ge 1$. We summarize the steps of finding the conditional expected waiting times of the k^{th} low priority customer, k=1,2,...,K below.

STEP 1) [Assign the conditional expected waiting times to 0 at all the infeasible states.]

$EW_M^{\ell}(n,0,M+1;k) \leftarrow 0$, n=R,...,N-1 and k=1,...,min(M,K)

$EW_M^{\ell}(n,0,m;0) \leftarrow 0$, n=R,...,N-1 and m=0,1,...,M

$EW_M^{\ell}(N,q,m;1) \leftarrow 0$, $q=0,1,\ldots,Q_1$ and m=0,1,...,M

STEP 2) [Compute the conditional expected waiting times for the k^{th} low priority customer.]

$k \leftarrow 1$

STEP 3) [Set up the starting states for the k^{th} low priority.]

$$EW_M^{\ell}(N,q,m;k) \leftarrow \frac{q+k}{N\mu-\lambda_1} \quad , \quad \begin{matrix} q=0,1,\ldots,Q_1 \\ m=M+k,\ldots,M+K \end{matrix}$$

STEP 4) $q_2 \leftarrow M + k - 1$

STEP 5) [Compute $EW_M^{\ell}(N,q,q_2;k)$ for q_2>M.]

If q_2>M and $q_2\ge k$ then define C_q and D_q, $q=0,1,\ldots,Q_1$

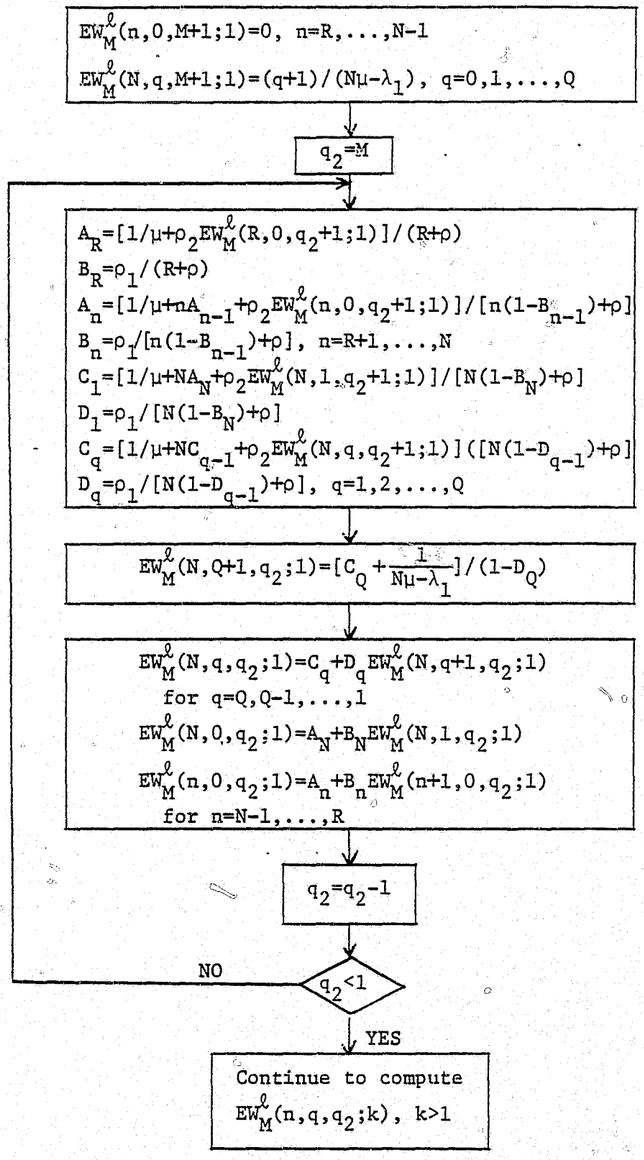


Exhibit 3.8 Block diagram of finding the conditional expected waiting times of the lst low priority customer

$$C_0 = \frac{1/\mu+NEW_M^{\ell}(N,0,q_2-1;k-1)+\rho_2 EW_M^{\ell}(N,0,q_2+1;k)}{N+\rho}$$

$$D_0 = \frac{\rho_1}{N+\rho}$$

$$C_q = \frac{1/\mu+NC_{q-1}+\rho_2 EW_M^{\ell}(N,q,q_2+1;k)}{N(1-D_{q-1})+\rho} \quad , \quad q=1,2,\ldots,Q_1$$

$$D_q = \frac{\rho_1}{N(1-D_{q-1})+\rho} \quad , \quad q=1,2,\ldots,Q_1$$

$$EW_M^{\ell}(N,Q_1+1,q_2;k) \leftarrow \frac{C_{Q_1}+\frac{1}{N\mu-\lambda_1}}{1-D_{Q_1}}$$

$$EW_M^{\ell}(N,q,q_2;k) \leftarrow C_q + D_q EW_M^{\ell}(N,q+1,q_2;k) \quad , \quad q=Q_1,Q_1-1,\ldots,0$$

$q_2 \leftarrow q_2 - 1$, and repeat this step.

If $q_2>M$ and $q_2<k$, GO TO STEP 7).

If $q_2 \leq M$, GO TO STEP 6).

STEP 6) [Compute $EW_M^{\ell}(n,q,q_2;k)$ for $q_2 \leq M$.]

$$A_R = \frac{1/\mu+REW_M^{\ell}(R,0,q_2-1;k-1)+\rho_2 EW_M^{\ell}(R+1,0,q_2;k-1)}{R+\rho} \quad \text{if } q_2=M$$

$$A_R = \frac{1/\mu+REW_M^{\ell}(R,0,q_2-1;k-1)+\rho_2 EW_M^{\ell}(R,0,q_2+1;k)}{R+\rho} \quad \text{if } q_2<M$$

$$B_R = \rho_1/(R+\rho)$$

$$A_n = \frac{1/\mu+nA_{n-1}+\rho_2 EW_M^{\ell}(n+1,0,q_2;k-1)}{n(1-B_{n-1})+\rho} \quad , \quad n=R+1,\ldots,N-1 \text{ when } q_2=M$$

$$A_n = \frac{1/\mu+nA_{n-1}+\rho_2 EW_M^{\ell}(n,0,q_2+1;k)}{n(1-B_{n-1})+\rho} \quad , \quad n=R+1,\ldots,N \text{ when } q_2<M \text{ and } n=N \text{ when } q_2=M$$

$$B_n = \rho_1/[n(1-B_{n-1})+\rho] \quad , \quad n=R+1,\ldots,N$$

$$C_1 = \frac{1/\mu+NA_N+\rho_2 EW_M^{\ell}(N,1,q_2+1;k)}{N(1-B_N)+\rho}$$

$$D_1 = \rho_1/[N(1-B_N)+\rho]$$

$$C_q = \frac{1/\mu+NC_{q-1}+\rho_2 EW_M^{\ell}(N,q,q_2+1;k)}{N(1-D_{q-1})+\rho} \quad , \quad q=2,3,\ldots,Q_1$$

$$D_q = \rho_1/[N(1-D_{q-1})+\rho] \quad , \quad q=2,3,\ldots,Q_1$$

$$EW_M^{\ell}(N,Q_1+1,q_2;k) \leftarrow \frac{C_{Q_1} + \frac{1}{N\mu-\lambda_1}}{1-D_{Q_1}}$$

$$EW_M^{\ell}(N,q,q_2;k) \leftarrow C_q + D_q EW_M^{\ell}(N,q+1,q_2;k) \quad , \quad q_2=Q_1,Q_1-1,\ldots,1$$

$$EW_M^{\ell}(N,0,q_2;k) \leftarrow A_N + B_N EW_M^{\ell}(N,1,q_2;k)$$

$$EW_M^{\ell}(n,0,q_2;k) \leftarrow A_n + B_n EW_M^{\ell}(n+1,0,q_2;k) \quad , \quad n=N-1,\ldots,R$$

$$q_2 \leftarrow q_2 - 1$$

If $q_2 \geq k$, repeat STEP 6). O.W. GO TO STEP 7).

STEP 7) $k \leftarrow k+1$

If $k \leq K$, GO TO STEP 3). O.W. STOP.

In Steps 5) and 6) of the above algorithm we used an approximation $EW_M^{\ell}(N,Q_1,q_2;k) \cong EW_M^{\ell}(N,Q_1+1,q_2;k) \cong \frac{1}{N\mu-\lambda_1}$, $q_2 \geq k \geq 1$. We state this in the following Theorem.

Theorem 3.2. $\lim_{q_1 \to \infty} [EW_M^{\ell}(N,q_1+1,q_2;k) - EW_M^{\ell}(N,q_1,q_2;k)] = \frac{1}{N\mu-\lambda_1}$

for $q_2 \geq k \geq 1$.

We are not presenting the proof here, but we will prove it in the dissertation. Note that Theorem 3.2 is a generalization of Lemma 3.5 for $k \geq 1$. The physical meaning of Theorem 3.2 is that, for the k^{th} low priority customer, the difference of the expected waiting time in the queue at state (N,q_1+1,q_2) and at state (N,q_1,q_2) approaches to a constant, $1/(N\mu-\lambda_1)$, when q_1 approaches to infinite, for every $q_2 \geq k \geq 1$. And, this constant is just the expected time of the system goes from state (N,q_1+1,q_2) to (N,q_1,q_2).

In this algorithm, the number of operations is of the order of $O(KM(N-R+1+Q_1))$, where K is the number of low priority customer one wants to compute, M is the cutoff point on the low priority queue, N-R is the number of servers reserved for high priority customer and Q_1 is an estimate of the maximum high priority queue length. In most of these models, $Q_1=5$ will give the very good approximations. We do a single example with $Q_1=2$ below.

Example 3.1. Find the conditional expected waiting times of the first 3 low priority customer in D(5;3,2;Q,Q) model with $\lambda_1=\lambda_2=\mu=1$.

Solution: In this example, we have N=5, R=3, M=2, K=3, $\rho_1=\rho_2=\mu=1$ and pick $Q_1=2$.

STEP 1) $EW_2^{\ell}(n,0,3;k) = 0$, n=3,4 and k=1,2

$EW_2^{\ell}(n,0,m;0) = 0$, n=3,4 and m=0,1,2

$EW_2^{\ell}(5,q_1,m;0) = 0$, q_1=0,1,2 and m=0,1,2

STEP 2) $k = 1$

STEP 3) $EW_2^{\ell}(5,q_1,m;1) = \frac{q_1+1}{4}$, q_1=0,1,2 and m=3,4,5

STEP 4) $q_2 = 2$

STEP 5) Skipped because $q_2 = M$.

STEP 6) $A_3 = .2$, $B_3 = .2$

$A_4 = .346$, $B_4 = .192$

$A_5 = .494$, $B_5 = .166$

$C_1 = .643$, $D_1 = .162$

$C_2 = .802$, $D_2 = .162$

$EW_2^{\ell}(5,3,2;1) = 1.255$

$EW_2^{\ell}(5,2,2;1) = 1.005$

$EW_2^{\ell}(5,1,2;1) = .806$

$EW_2^{\ell}(5,0,2;1) = .627$

$EW_2^{\ell}(4,0,2;1) = .467$

$EW_2^{\ell}(3,0,2;1) = .293$

$q_2 = 2-1 = 1 \geq k = 1$ (repeat STEP 6)

STEP 6) $A_3 = .259$, $B_3 = .2$

$A_4 = .481$, $B_4 = .192$

$A_5 = .668$, $B_5 = .166$

$C_1 = .834$, $D_1 = .162$

$C_2 = .997$, $D_2 = .162$

$EW_2^{\ell}(5,3,1;1) = 1.494$

$EW_2^{\ell}(5,2,1;1) = 1.238$

$EW_2^{\ell}(5,1,1;1) = 1.034$

$EW_2^{\ell}(5,0,1;1) = .839$

$EW_2^{\ell}(4,0,1;1) = .642$

$EW_2^{\ell}(3,0,1;1) = .387$

$q_2 = 1-1 = 0 < k = 1$, GO TO STEP 7).

STEP 7) $k = 1+1 = 2 \leq K = 3$, GO TO STEP 3).

STEP 3) $EW_2^{\ell}(5,q,m;2) = \frac{q+2}{4}$, q=0,1,2 and m=4,5.

STEP 4) $q_2 = 3$

STEP 5) $q_2 = 3 > M = 2$ and $q_2 = 3 \geq 2$

$C_0 = .662$, $D_0 = .143$

$C_1 = .805$, $D_1 = .159$

$C_2 = .971$, $D_2 = .161$

$EW_2^{\ell}(5,3,3;2) = 1.457$

$EW_2^{\ell}(5,2,3;2) = 1.206$

$EW_2^{\ell}(5,1,3;2) = .997$

$EW_2^{\ell}(5,0,3;2) = .805$

$q_2 = 3-1 = 2$, and repeat STEP 5).

STEP 5) $q_2 = 2 \leq M$, GO TO STEP 6)

STEP 6) $A_3 = .526$, $B_3 = .2$

$A_4 = .717$, $B_4 = .192$

$A_5 = .893$, $B_5 = .166$

$C_1 = 1.047$, $D_1 = .162$

$C_2 = 1.202$, $D_2 = .162$

$EW_2^{\ell}(5,3,2;2) = 1.734$

$EW_2^{\ell}(5,2,2;2) = 1.482$

$EW_2^{\ell}(5,1,2;2) = 1.287$

$EW_2^{\ell}(5,0,2;2) = 1.106$

$EW_2^{\ell}(4,0,2;2) = .93$

$EW_2^{\ell}(3,0,2;2) = .712$

$q_2 = 2-1 = 1 < k = 2$, GO TO STEP 7).

STEP 7) $k = 2+1 = 3 \leq K = 3$, GO TO STEP 3).

STEP 3) $EW_2^{\ell}(5,q,m;3) = \frac{q+3}{4}$, $q=0,1,2$, $m=5$

STEP 4) $q_2 = 4$

STEP 5) $q_2 = 4 > M = 2$ and $q_2 = 4 \geq k = 3$

$C_0 = .825$, $D_0 = .143$

$C_1 = .974$, $D_1 = .159$

$C_2 = 1.148$, $D_2 = .161$

$EW_2^{\ell}(5,3,4;3) = 1.667$

$EW_2^{\ell}(5,2,4;3) = 1.416$

$EW_2^{\ell}(5,1,4;3) = 1.2$

$EW_2^{\ell}(5,0,4;3) = .996$

$q_2 = 4-1 = 3$, and repeat STEP 5).

STEP 5) $q_2 = 3 > M = 2$ and $q_2 = 3 \geq k = 3$

$C_0 = 1.075$, $D_0 = .143$

$C_1 = 1.205$, $D_1 = .159$

$C_2 = 1.361$, $D_2 = .161$

$EW_2^{\ell}(5,3,3;3) = 1.92$

$EW_2^{\ell}(5,2,3;3) = 1.67$

$EW_2^{\ell}(5,1,3;3) = 1.471$

$EW_2^{\ell}(5,0,3;3) = 1.285$

$q_2 = 3-1 = 2$, repeat STEP 5).

STEP 5) $q_2 = 2 \leq M$ and $q_2 = 2 < k = 3$, GO TO STEP 7).

STEP 7) $k = 3+1 = 4 > K = 3$, STOP.

We summarize the results below in Exhibit 3.9. The numbers below each state are the expected waiting times for each waiting low priority customer.

(3,0,1) .387	(3,0,2) .293, .712	
(4,0,1) .642	(4,0,2) .467, .93	
(5,0,1) .839	(5,0,2) .627, 1.106	(5,0,3) .25, .805, 1,285
(5,1,1) 1.034	(5,1,2) .806, 1.287	(5,1,3) .5, .997, 1.471
(5,2,1) 1.238	(5,2,2) 1.01, 1.482	(5,2,3) .75, 1.206, 1.67

Exhibit 3.9 Conditional expected waiting times of low priority customer in D(5;3,2;Q,Q) model

As we can see from Exhibit 3.9, the numbers are slightly greater than those numbers in the corresponding states of D(5;3,2;L,Q) model (see page 45).

Following two properties which hold in the D(N;R,M;L,Q) model also hold in the D(N;R,M;Q,Q) model.

(1) The busier the system or (when the system is full) the more high priority customers waiting, the greater the conditional expected waiting time for low priority customer. That is,

$$EW_M^{\ell}(R,0,q_2;k)<\ldots<EW_M^{\ell}(N,0,q_2;k)<\ldots<EW_M^{\ell}(N,q_1,q_2;k)<EW_M^{\ell}(N,q_1+1,q_2;k)$$

for $q_2 \geq k \geq 1$, $q_1>0$.

(2) The more low priority customer waiting, the less conditional expected waiting time of low priority customer in the same position. That is,

for $R \le n < N$, $EW_M^\ell(n,0,k;k) > EW_M^\ell(n,0,k+1;k) > \ldots > EW_M^\ell(n,0,M;k)$, $1 \le k < M$

for $n=N$, $EW_M^\ell(N,q_1,k;k) > \ldots > EW_M^\ell(N,q_1,M;k) > EW_M^\ell(N,q_1,q_2;k)$, $q_2 > M > k$, $q_1 = 0,1,2,\ldots$

The reason for above two properties are the same as they are in the D(N;R,M;L,Q) model.

Let EW_M^ℓ be the unconditional expected waiting time for the D(N;R,M;Q,Q) model. Then

$$EW_M^\ell = \sum_{q_2=0}^{M-1} \sum_{n=R}^{N-1} P(n,0,q_2)\ EW_M^\ell(n,0,q_2+1;q_2+1)$$

$$+ \sum_{n=R}^{N-1} P(n,0,M)\ EW_M^\ell(n+1,0,M;M)$$

$$+ \sum_{q_2=0}^{\infty} \sum_{q_1=0}^{\infty} P(N,q_1,q_2)\ EW_M^\ell(N,q_1,q_2+1;q_2+1) \qquad (3.56)$$

The reason for (3.56) is that: if a low priority customer arrives and finds 1) the system at state $(n,0,q_2)$, $R \le n < N$ and $q_2 < M$, then he/she will join the low priority queue in the $(q_2+1)^{st}$ position with the expected waiting time equals $EW_M^\ell(n,0,q_2+1;q_2+1)$, 2) the system at state $(n,0,M)$, $R \le n < N$, then the first low priority customer in the queue will begin the service and the arriving low priority customer will join the queue in the M^{th} position with the expected waiting time equals $EW_M^\ell(n+1,0,M;M)$, 3) the system at state (N,q_1,q_2), then he/she will join the low priority queue in the $(q_2+1)^{st}$ position with the expected waiting time equals $EW_M^\ell(N,q_1,q_2+1;q_2+1)$.

A program has been written to compute $P(n,q_1,q_2)$, $EW_M^\ell(n,q_1,q_2;k)$ and using (3.56) to compute EW_M^ℓ with q_1 up to Q_1 and q_2 up to Q_2. For R=N-1, the EW_M^ℓ should be bounded below by the EW_M^ℓ, denoted as EW_0^ℓ in Exhibit 3.10, in the D(N;N,∞;Q,Q) model and bounded above by the EW_M^ℓ, denoted as EW_∞^ℓ in Exhibit 3.10, in the D(N;N-1,∞;Q,Q) model. And, $EW_M^\ell \to EW_\infty^\ell$ as $M \to \infty$. Exhibit 3.10 verifies this.

N	R	λ_1	λ_2	μ	M	EW_M^ℓ
5	4	1	2	1	0	.0249
5	4	1	2	1	1	.0614
5	4	1	2	1	5	.0867
5	4	1	2	1	10	.0876
5	4	1	2	1	20	.0876
5	4	1	2	1	∞	.0876
10	9	2	4	1	0	.0317
10	9	2	4	1	1	.0448
10	9	2	4	1	5	.0656
10	9	2	4	1	10	.0704
10	9	2	4	1	20	.0712
10	9	2	4	1	∞	.0712
15	14	.05	.25	.034	0	.2196
15	14	.05	.25	.034	1	.3052
15	14	.05	.25	.034	5	.4394
15	14	.05	.25	.034	10	.4676
15	14	.05	.25	.034	20	.4714
15	14	.05	.25	.034	∞	.4715

Exhibit 3.10 Comparisons of the unconditional expected waiting time of low priority customer in D(N;N-1,M;Q,Q) model

Now we discuss the bounds of $EW_M^\ell(n,q_1,q_2;k)$ in D(N;R,M;Q,Q) model. For any M, $EW_M^\ell(n,q_1,q_2;k)$ is bounded above by $EW_\infty^\ell(n,q_1,q_2;k)$ which we discussed earlier in Section 3.2. That is

$$EW_M^{\ell}(n,q_1,q_2;k) < EW_\infty^{\ell}(n,q_1,q_2;k) \ , \quad k \le q_2$$

at any corresponding feasible state of D(N;R,M;Q,Q) model and D(N;R,∞;Q,Q) model. The reasons for this is the same as it is for the D(N;R,M;L,Q) model. For example, the state (12,0,1) is a feasible state for D(15;12,∞;Q,Q) model and D(15;12,M;Q,Q) model for every M. With λ_1=.05, λ_2=.25, μ=.034, $EW_M^{\ell}(12,0,1;1)$ are displayed in the following table for M=1,..,7 and ∞.

M	1	2	3	4	5	6	7	∞
$EW_M(12,0,1;1)$	1.59	2.25	2.52	2.65	2.71	2.74	2.75	2.76

As we can see from the above table, it does not take too long to reach the upper bound.

As it is in the D(N;R,M;L,Q) model, $EW_M^{\ell}(n,q_1,q_2;k)$ is bounded below by $EW_M^{\ell}(n,0,M;k)$ for $R \le n \le N$, $k \le q_2 \le M$ and the lower bound is exact when q_2=M. That is,

$$EW_M^{\ell}(n,q_1,M;k) < EW_M^{\ell}(n,q_1,M-1;k) < \ldots < EW_M^{\ell}(n,q_1,q_2;k) < \ldots < EW_M^{\ell}(n,q_1,k;k)$$

for n=R,...,N and $1 \le k < M$.

To find the lower bound $EW_M^{\ell}(n,q_1,M;k)$, we have to carry out the recursive sequences $A_n, B_n, C_{q_1}, D_{q_1}$ which are defined in this section. It will be easier to find the lower bound than the exact value. We do a simple example below to show how to find the bounds.

Example 3.2. In the D(4;3,3;Q,Q) model with $\lambda_1=\lambda_2=\mu=1$. A low priority customer arrives and find the system is at state (3,0). What is the maximum expected waiting time and what is the minimum expected waiting time of this low priority customer?

Solution. The arriving low priority customer will enter the low priority queue in the first position and the expected waiting time for him/her is $EW_3^{\ell}(3,0,1;1)$. From above discussion we have

$$EW_3^{\ell}(3,0,3;1) < EW_3^{\ell}(3,0,1;1) < EW_\infty^{\ell}(3,0,1;1)$$

(1) Compute the upper bound $EW_\infty^{\ell}(3,0,1;1)$.

From Section 3.2 we have

$$D_N = \frac{1}{N\mu-\lambda_1}$$

$$D_{N-1} = D_R = \frac{1+\lambda_1 D_N}{R\mu} = \frac{N\mu}{R\mu(N\mu-\lambda_1)} = \frac{4}{9} = .444$$

$$EW_\infty^{\ell}(3,0,1;1) = .444$$

(2) Compute the lower bound $EW_3^{\ell}(3,0,3;1)$

For simplicity, we pick Q_1=2. Then we have

$A_3 = .2$, $B_3 = .2$

$A_4 = .4103$, $B_4 = .1923$

$C_1 = .6324$, $D_1 = .1912$

$C_2 = .8652$, $D_2 = .191$

$EW_3^{\ell}(3,0,3;1) = .3148$

Hence, we have $.3148 < EW_3^{L}(3,0,1;1) < .4444$

In general, for a given model D(N;R,M;Q,Q), it is much easier to find the upper bound than the lower bound, numerically.

We can define the boundary states in the D(N;R,M;Q,Q) model in the same way as they are in the D(N;R,M;L,Q) model. All the properties about the boundary states also hold in the D(N;R,M;Q,Q) model (see page 51) and we do not repeat here.

Now we show the example we did in Section 2.3, the D(15;12,5;L,Q) model. When the system is full, instead of lost, the high priority calls are queued

in the high priority queue. Hence, it is D(15;12,5;Q,Q) model. Exhibit 3.11 is the transition diagram for D(15;12,5;Q,Q) model. Exhibits 3.12 and 3.13 are the steady state probabilities and the conditional expected waiting times of low priority calls, respectively, for those states that are in the dashed box in Exhibit 3.11. Comparing the steady state probability in the D(15;12,5;Q,Q) model to the steady state probability in D(15;12,5;L,Q) model, one can notice that the former's probability is slightly less than the latter's. As a matter of fact, this is true in general because in the D(N;R,M;Q,Q) model it has more states than the D(N;R,M;L,Q) model has.

The properties about the conditional expected waiting times of low priority calls in the D(N;R,M;L,Q) model also hold in the D(N;R,M;Q,Q) model. Comparing the conditional expected waiting time of low priority call in D(15;12,5;Q,Q) model to the conditional expected waiting time of low priority call in D(15;12,5;L,Q) model, one can notice that the former is slightly greater than the latter. This is also true in general because in D(N;R,M;Q,Q) model there is a probability that high priority calls may come into the system when the system is full and those high priority calls will be served before any of the waiting low priority calls can go into the service. The unconditional expected waiting time of low priority call also has this property: In D(15;12,5;Q,Q) model, the unconditional expected waiting time of low priority call is 1.705 and in D(15;12,5;L,Q) model, the unconditional expected waiting time of low priority is 1.66.

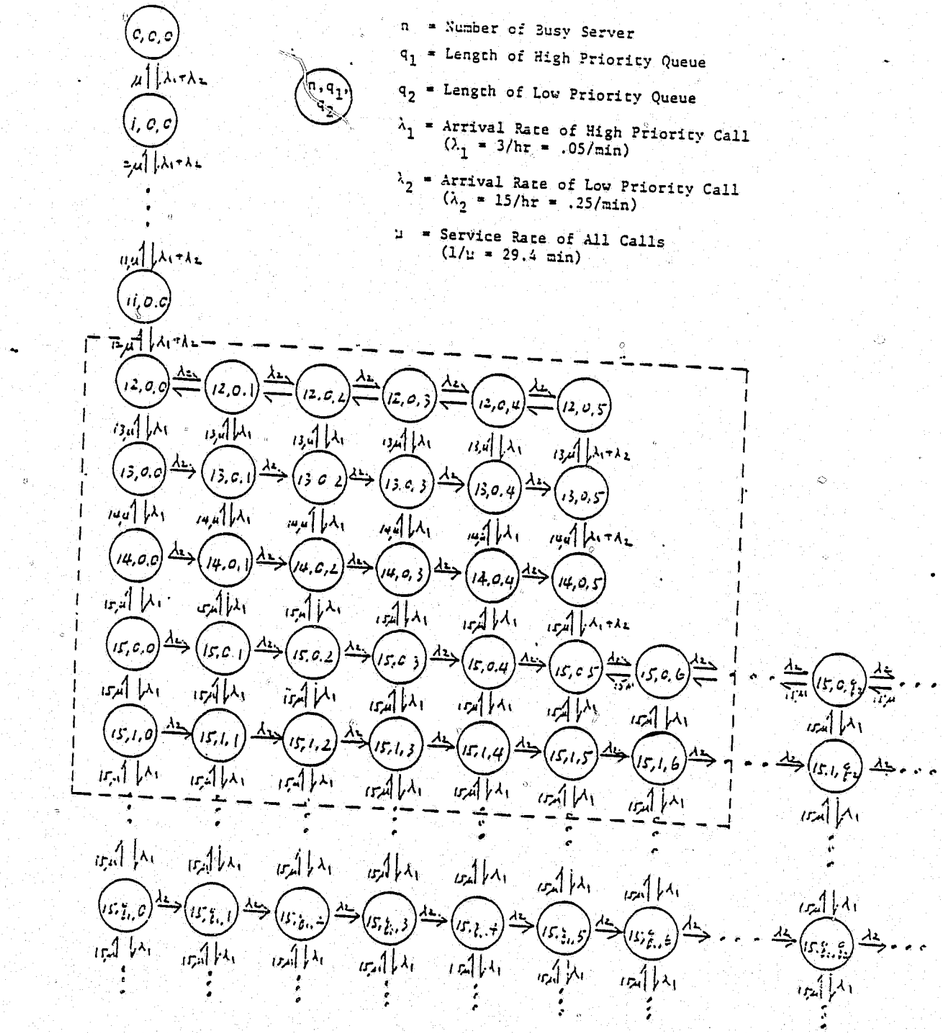


Exhibit 3.11 Transition diagram for D(15;12,5;Q,Q) model

THE STEADY STATE PROB FOR D(15;12, 5;Q,Q)

LAMDA1= 0.0500, LAMDA2= 0.2500, MU= 0.0340

THE PROB OF HIGH PRIORITY CUSTOMER WAIT IS 0.00802

THE PROB OF LOW PRIORITY CUSTOMER WAIT IS 0.22418

12, 0, 0	13, 0, 0	14, 0, 0	15, 0, 0	15, 1, 0
.6478E-01	.4562E-02	.3069E-03	.1975E-04	.1270E-05
12, 0, 1	13, 0, 1	14, 0, 1	15, 0, 1	15, 1, 1
.4269E-01	.4685E-02	.4230E-03	.3384E-04	.2604E-05
12, 0, 2	13, 0, 2	14, 0, 2	15, 0, 2	15, 1, 2
.2931E-01	.3814E-02	.4068E-03	.3763E-04	.3302E-05
12, 0, 3	13, 0, 3	14, 0, 3	15, 0, 3	15, 1, 3
.2057E-01	.2889E-02	.3400E-03	.3472E-04	.3358E-05
12, 0, 4	13, 0, 4	14, 0, 4	15, 0, 4	15, 1, 4
.1461E-01	.2128E-02	.2657E-03	.2902E-04	.3017E-05
12, 0, 5	13, 0, 5	14, 0, 5	15, 0, 5	15, 1, 5
.1044E-01	.8456E-02	.5486E-02	.3243E-02	.2097E-03
			15, 0, 6	15, 1, 6
			.1700E-02	.1798E-03

Exhibit 3.12 Steady state probabilities for D(15;12,5;Q,Q) model

The Conditional Expected Waiting Times In D(15;12, 5;0,0)

LAMDA1= 0.0500, LAMDA2= 0.2500, MU= 0.0340

The unconditional EW = 1.7050. The EWC = 7.6058

[12, 0, 1] 2.71	[13, 0, 1] 5.11	[14, 0, 1] 7.22	[15, 0, 1] 9.08	[15, 1, 1] 10.84
[12, 0, 2] 2.65 5.31	[13, 0, 2] 4.93 7.57	[14, 0, 2] 6.87 9.51	[15, 0, 2] 8.53 11.22	[15, 1, 2] 10.08 12.83
[12, 0, 3] 2.53 5.11 7.75	[13, 0, 3] 4.58 7.16 9.81	[14, 0, 3] 6.22 8.80 11.58	[15, 0, 3] 7.61 10.37 13.14	[15, 1, 3] 8.95 11.91 14.64
[12, 0, 4] 2.25 4.72 7.31 9.96	[13, 0, 4] 3.87 6.43 9.11 11.82	[14, 0, 4] 5.06 7.85 10.64 13.41	[15, 0, 4] 6.17 9.14 12.02 14.85	[15, 1, 4] 7.42 10.46 13.40 16.27
[12, 0, 5] 1.59 3.96 6.52 9.20 11.92	[13, 0, 5] 2.50 5.17 7.98 10.78 13.56	[14, 0, 5] 3.09 6.30 9.28 12.17 15.01	[15, 0, 5] 4.20 7.50 10.56 13.50 16.38	[15, 1, 5] 5.68 8.89 11.93 14.69 17.78
			[15, 0, 6] 2.17 5.68 8.89 11.93 14.89 17.78	[15, 1, 6] 4.35 7.40 10.44 13.42 16.35 19.24

Exhibit 3.13 Conditional expected waiting times of low priority customer for D(15;12,5;Q,Q) model

REFERENCES

1. Benn, B.A., "Hierarchical Car Pool System in Railroad Transportation", Ph.D. Dissertation, Case Institute of Technology, 1967. (University Microfilms)

2. Cahn, M.F. and Tien, J.M., An Alternative Approach In Police Response: The Wilmington Management Of Demand Program, Cambridge, MA: Public Systems Evaluation, Inc., March 1981.

3. Chaiken, J.M., "A Patrol Car Allocation Model: Background", Management Science, 24, pp. 1280-1290, 1978.

4. Chaiken, J.M., "A Patrol Car Allocation Model: Capabilities and Algorithms", Management Science, 24, pp. 1291-1300, 1978.

5. Cobham, A., "Priority Assignment in Waiting Line Problems", Operations Research, 2, pp. 70-76, 1954.

6. Cooper, R.B., Introduction To Queueing Theory, New York, NY: North Holland, 1981.

7. Davis, R.H., "Waiting Time Distribution of a Multi-Server Priority Queuing System", Operations Research, 14, pp. 133-136, 1966.

8. Dressin, S.A. and Reich, E., "Priority Assignment on a Waiting Line", Quart. Appl. Math., 15, pp. 208-211, 1957.

9. Esogbue, A.O. and Singh, A.J., "A Stochastic Model for an Optimal Priority Bed Distribution Problem in a Hospital Ward", Operations Research, 24, pp. 884-898, 1976.

10. Feller, W., An Introduction To Probability Theory And Its Applications, Vol. 2, Second Edition, New York, NY: John Wiley & Sons, Inc., 1975.

11. Gauer, D.P., Jr., "Diffusion Approximations and Models for Certain Congestion Problems", Journal of Applied Probability, 5, pp. 607-623, 1968.

12. Green, L., "Queues Which Allow A Random Number Of Servers Per Customer", Ph.D Dissertation, Yale University, 1978.

13. Green, L., "A Queueing System In Which Customers Require A Random Number of Servers", Operations Research, Vol. 28, pp. 1335-1346, 1980.

14. Green, L., "Comparing Operation Characteristics Of Queues In Which Customers Require A Random Number of Servers", Management Science, Vol. 27, pp. 65-74, 1981.

15. Gross, D. and Harris, C.M., Fundamentals Of Queuing Theory, New York, NY: John Wiley & Sons, Inc., 1974.

16. Jaiswal, N.K., Priority Queues, New York, NY: Academic Press, 1968.

17. Kansas City Police Department, Response Time Analysis, Washington, DC: U.S. Government Printing Office, 1977.

18. Kendall, M.G., and Stuart, A., The Advance Theory Of Statistics, Vol. 2, Second Edition, London: Griffin, 1963.

19. Kleinrock, L., Queuing System, Vol. 1, 2, New York, NY: John Wiley & Sons, Inc., 1975.

20. Larson, R.C., Urban Police Patrol Analysis, Cambridge, MA: The MIT Press, 1972.

21. Larson, R.C., Hypercube Queuing Model, New York, NY: The New York City Rand Institute, R-168812-HUD, July 1975.

22. Morse, P.M., Queues, Inventories And Maintenance, New York, NY: John Wiley & Sons, Inc., 1967.

23. Newell, G.F., "Approximate Methods for Queues With Application To The Fixed-Cycle Traffic Light", SIAM Review, 7, pp. 223-240, 1965.

24. Newell, G.F., "Queues With Time-Dependent Arrival Rates I-III", Journal of Applied Probability, 5, pp. 436-451, 579-606, 1968.

25. Newell, G.F., Applications of Queuing Theory, London: Chapman and Hall, 1971.

26. Parzen, E., Stochastic Processes, San Francisco: Holden-Day, 1962.

27. Shonick, W. and Jackson, J.R., "An Improved Stochastic Model For Occupancy-Related Random Variables In General-Acute Hospitals", Operations Research, 21, pp. 952-965, 1973.

28. Sumrall, R.O. et al., Alternative Strategies For Responding To Police Calls For Service, Birmingham, Alabama: Birmingham Police Department, Unpublished Report, 1980.

29. Taylor, I.D.S., "A Priority Queuing Model To Measure The Performance In The Ontario Ambulance System", Ph.D Dissertation, University of Toronto, 1976.

30. Taylor, I.D.S and Templeton, J.G.C., "Multi-Server Priority Queues With Modified Cutoff Queue Discipline", University of Toronto, Department of Industrial Engineering, Working Paper #76-018, October 1976.

31. Taylor, I.D.S. and Templeton, J.G.C., "Waiting Time In A Multi-Server Cutoff Priority Queue And Its Application To An Urban Ambulance Service", Operations Research, 28, pp. 1168-1188, 1980.

32. Tien, J.M., "Control Of A Two Customer Class Interactive-Multi-Facility Queuing System", Technical Report No. 73, Operations Research Center, Massachusetts Institute of Technology, June 1972.

33. Tien, J.M., Simon, J.W. and Larson, R.C., An Alternative Approach In Police Patrol: The Wilmington Split-Force Experiment, Washington, DC: U.S. Government Printing Office, No. 027-000-0068-0, April 1978.

34. Tien, J.M. and Valiante, N.M., "A Case For Formally Delaying Non-Critical Calls For Service", The Police Chief, March, 1979.

35. Tien, J.M. and Colton, K.W., "Police Command, Control And Communications", in How Well Does It Work? Review Of Criminal Justice Evaluation, 1978, Washington, DC: U.S. Government Printing Office, No. 027-000-0082-8, June 1979.

36. Yee, J.R., "On The First Passage Times In M/M/1/K And M/M/S/K Queuing System", Electrical, Computer, and Systems Engineering Dept., Rensselaer Polytechnic Institute, 1981.

END