Exploring a New Markov Chain Model for Multiqeue Systems

G. Mapp, D. Thakker and O. Gemikonakli

School of Engineering and Information Sciences
Middlesex University, Hendon Campus
London, UK, NW4 4BT
Email: g.mapp, d.thakker, o.gemikonakli@mdx.ac.uk

***Abstract*—Traditionally, Markov models have been used to study multiserver systems using exhaustive or gated service. In addition, exhaustive-limited and gate-limited models have also been used in communication systems to reduce overall latency. Recently the authors have proposed a new Markov Chain approach to study gate-limited service. Multiqueue systems such as polling systems, in which the server serves various queues have also been extensively studied but as a separate branch of queueing theory. This paper proposes to describe multiqueue systems in terms of a new Markov Chain called the Zero-Server Markov Chain (ZSMC). The model is used to derive a formula for the waiting times in an exhaustive polling system. An intuitive result is obtained and this is used to develop an appoximate method which works well over normal operational ranges.**

***Keywords*-Polling Systems, Markov Models, Unification**

I. Introduction

Polling systems, where a system of multiple queues are served in cyclic order by a single server, have been intensely studied. The original impetus came from the operations research area, looking at maximizing the use of machinery such as the patrolling machine repairman problem in the $1950s$ [1]. This was pursued in the $60s$ and $70s$ with the advent of central computing facilities where a central server polled data from terminals distributed around large facilities such as univerity campuses. However, with the advent of computer networks in the $80s$, these models were also used to study slotted and token rings [2] [3].

Traditionally, two types of service disciplines and their variants have been studied. The first is called **exhaustive** service where the server serves everyone in the queue until the queue is empty. This means that customers arriving at the queue while the server is serving at the queue are served during the current service period [4]. The second is called **gated** service, where the server only serves the customers it finds in the queue at the beginning of the service period for that queue. Customers arriving during the current service period must therefore be serviced during subsequent server visits to that queue. In exhaustive-limited and gated-limited systems, the relevant discipline is followed but only a maximum number of customers, denoted by K, are served in any one visit to the queue. Though we have exact solutions for exhaustive and gated service [5], [6], solutions for exhaustive-limited and gated-limited system have been more difficult to obtain, with numerical soultions requiring large amounts of computational power as the number of queues increases. Recent efforts have therefore been focused on getting faster algorithms to compute the waitng times in these systems.

The study of multiserver systems, in which a number of servers simultaneously serve the same queue, intensified within the computer community with the development of multiprocessors systems in the $90s$ [7]. With the advent of mobile communications [8], multiserver systems have become more closely studied. For multiserver systems with a limited number of servers, say K servers, the deployment of servers can also be classified as exhaustive-limited or gate-limited. In exhaustive-service, all K servers are always deployed and so a customer can enter service at an empty server while other customers are being served by other servers. This is also called **Partial Batch Service**. An example of this kind of service is a wireless network such as 3G [9] and other mobile systems, where channels are allocated in a dynamic manner.

In gated-limited service, if the number of customers in the queue at the end of the service period is less than the maximum K, say m, then only m servers are assigned for this service period. Customers arriving after service has begun must be served in subsequent cycles. An example of such a system is a network-based service in which applications can post requests using a network buffer. The buffer is then sent over to the server to satisfy requests. New requests that arrive after the buffer has been sent, must wait until replies to previous requests have been returned to the client machine, hence this can be regarded as a gated-limited multiserver system.

It is fair to say that the study of multiserver systems and multiqueue systems have been mostly studied as separate systems. Multiqueue or polling systems tend to be studied by looking at scan and departure instants of the server while multiserver systems tend to be analsyed at the end of different service periods. In this paper we propose a new Markov chain analysis to unify both system types. The rest of the paper is structured as follows: In Section 2, traditional techniques for solving multiqueue systems are described, while Section 3 looks at traditional multiserver solutions. In Section 4, the new Markov Model for multiserver gate-limited service is presented and the general solution is outlined. In Section 5, we show how this model can be extended to examine multiqueue

systems by introducing the concept of a Zero-Server Markov Chain. This concept is used to look at the Markov Model for a polling system with exhaustive service. A simple solution is obtained. In Section 6, an approximate method for calculating the average waiting time in a symmetrical polling system with exhaustive service is presented. The paper is concluded in Section 7.

II. Traditional Approach to Solving Polling Models

There have been several approaches used to analyse these systems [10]. Cyclic systems with probabilistically-limited service were examined in [11]. The authors in [12], explored pseudo-conservation laws to examine cyclic systems with several limited service policies, including gated-limited ones. In both cases, solutions were found but several iterations were required to yield a useable result. Recent work explored new algorithms to improve the accuracy and speed with moder ate sucess [13], [14]. In [15], the authors looked at using a Markov Chain Model for a polling system with parameter regeneration. In this system, the arrival and service rates may be changed every time a queue is exhaustively served. This approach allows external factors to be incorporated into polling systems.

III. Traditional Solutions for Multiserver Systems

A. Multiserver Exhaustive-Limted Service (Partial Batch Model)

We first examine the multiserver exhaustive-limited service or the Partial Batch Model (PBM) described in [16]. In this model a server can serve up to a maximum of K requests. If there are less than K requests in the system, the server begins to serve these requests. Furthermore, when there are less than K requests being serviced, new arrivals enter service until K requests are served or the queue is empty.

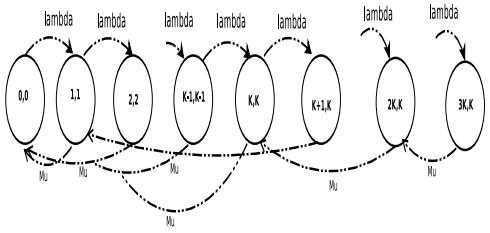

Fig. 1: Partial Batch Model.

This model is represented in Figure 1. Each state of the model is represented in terms of n and s where n is the total number of requests in the system and s is the number of requests being served. It can be seen from the figure that any new arrival enters service immediately as long as there are less than K number of requests being served. The time taken to service those requests is exponentially distributed to a mean value of $\frac{1}{\mu}$.

A stochastic balance equation for the model can be written as:

$$\begin{aligned} 0 &= -(\lambda+\mu)p_n + \mu p_{n+k} + \lambda p_{n-1} \\ 0 &= -\lambda p_0 + \mu p_1 + \mu p_2 + \mu p_{K-1} + \mu p_K \end{aligned} \tag{1}$$

This equation can be rewritten as:

$$[\mu D^{K+1} - (\lambda+\mu)D + \lambda]p_n = 0 \tag{2}$$

where p_n represents state probability and $n = 0, 1, 2...$ etc.

By finding the root r_0 of this equation that is between 0 and 1, one can work out the mean queue length (L) and average waiting time (W) for the queue, using the equations below:

$$L = \frac{r_0}{1-r_0} \quad \text{and } W = \frac{r_0}{\lambda(1-r_0)} \tag{3}$$

Note that for $K = 1$, we have the results for the M/M/1 queue, with $r_0 = \rho = \frac{\lambda}{\mu}$.

IV. A Multiserver Gated-Limited Model

In this section, we review a new Markov model for gated-limited multisever service. The model is based on Markov states represented by same two parameters used in the Partial Batch Model, n and s. For each cycle, we serve a maximum of K requests, so $s_{max} = K$. The model is shown in Figure 2. We start off with the empty state $0, 0$. If a request arrives, the system moves to state $1, 1$ as the request is immediately sent to the server. If more requests arrive before the server has returned, they are not served until the server returns. When the first request is served, the number of people served in the next cycle will depend on the number of requests in the queue when the service time has been completed. Thus for high instantaneous arrival rates, the system moves up the chains and for lower instantaneous arrival rates, it descends the chains.

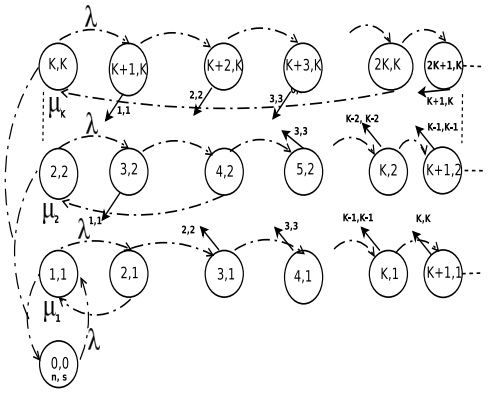

Fig. 2: Markov Chains for Gate-Limited Multiserver model.

In order to solve these equations, we need to be able to relate all the states of a chain, m, back to the first element of that chain. For $m < K$ that relationship which is represented by:

$$p_{n,m} = (\frac{\lambda}{(\lambda+\mu_m)})^{n-m} p_{m,m} \tag{4}$$

For $m = K$, we must solve for r_K using the same technique as in the Partial Batch Model.

A. Previous Results

Looking at the simple case of $K = 2$, we will have two chains which we will express in terms of the first elements in each chain:

So for Chain 1: we have

$$p_{n,1} \quad = (\tfrac{\lambda}{\lambda+\mu_1})^{n-1} p_{1,1} \tag{5}$$

For Chain 2: we have

$$p_{n,2} = r^{n-2} p_{2,2} \tag{6}$$

To completely describe the system we define the relationship between all the boundary states, i.e., $p_{0,0}$, $p_{1,1}$ and $p_{2,2}$.

$$\lambda p_{0,0} = \mu_1 p_{1,1} + \mu_2 p_{2,2} \tag{7}$$

$$(\lambda+\mu_1) p_{1,1} \quad = \lambda p_{0,0} + \mu_1 p_{2,1} + \mu_2 p_{3,2} \tag{8}$$

$$(\lambda+\mu_2) p_{2,2} \quad = \mu_2 p_{4,2} + \mu_1 p_{3,1} \tag{9}$$

We can now find the roots of these equations using the same technique as in the Partial Batch Model (PBM). Solution details are given in [17]. The results for $K = 2$ were compared with simulation and were shown to be fairly accurate over a wide operational range.

B. A General Solution

In this section, we seek to extend the method used for $K = 2$ to a general value of K. So a gate-limited model, where K is equal to the maximum number of requests that can be served at any moment, can be represented by a gated-limited model of K series or chains. Furthermore, if we represent a given chain by m, we can express the average number of requests in that chain, L_m, in terms of the first element of that chain, $p_{m,m}$. For $m < K$, this sum for that chain is given by:

$$L_m = \sum_{n=m}^{\infty} n (\frac{\lambda}{(\lambda+\mu_m)})^{n-m} p_{m,m} \tag{10}$$

Expanding:

$$L_m = \sum_{n=m}^{\infty} n - (m-1)(\frac{\lambda}{(\lambda+\mu_m)})^{n-m} p_{m,m} + (m-1) \sum_{n=m}^{\infty} (\frac{\lambda}{(\lambda+\mu_m)})^{n-m} p_{m,m} \tag{11}$$

Using the same technique as above and by letting $r_m = \frac{\lambda}{\lambda+\mu_m}$, we get the following solution:

$$L_m = \frac{m-(m-1)*r_m}{(1-r_m)^2} p_{m,m} \tag{12}$$

$$L = \sum_{m=1}^{K} L_m = \sum_{m=1}^{K} \frac{m-(m-1)*r_m}{(1-r_m)^2} p_{m,m} \tag{13}$$

For $m < K$,

$$r_m = \frac{\lambda}{\lambda+\mu_m} \tag{14}$$

For $m = K$, we use the imaginary PBM technique to solve for r_K. Furthermore, for $m < K$, we can sum the probabilities in each chain,

$$S_m = \sum_{n=m}^{\infty} (\frac{\lambda}{(\lambda+\mu_m)})^{n-m} p_{m,m} \tag{15}$$

Let $q = n - m$:

$$S_m = \sum_{q=0}^{\infty} (\frac{\lambda}{(\lambda+\mu_m)})^{q} p_{m,m}$$

$$S_m = \frac{\lambda+\mu_m}{\mu_m} p_{m,m} \tag{16}$$

If we let $p_{m,m} = C_{m,m} p_{K,K}$, we can express $p_{K,K}$ as:

$$p_{K,K} = \frac{1}{C_{0,0} + \sum_{m=1}^{m=K-1} \frac{\lambda+\mu_m}{\mu_m} C_{m,m} + \frac{1}{1-r_k}} \tag{17}$$

So to solve for any value of K we need to find the value of $C_{m,m}$ using the boundary equations for $p_{m,m}$. For $K = 2$, these equations are Equations 7, 8 and 9.

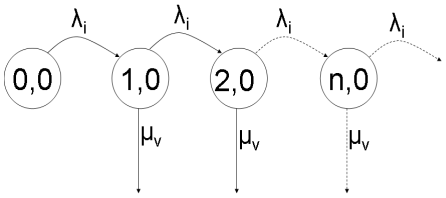

Fig. 3: A Zero-Server Markov Chain

V. Incorporating Multiqueue Systems into the Multiserver Model

In this section we seek to describe polling systems using the model of the set of Markov Chains used in the multiserver gate-limited described above. To do this we must first relax the restriction that the state of the queue in a polling system can only be satifactorily described at the scan and departure instances of the server, i.e., only when the server is present at the queue. We assert that the arrival and departure instances of customers are also valid observational points, just as in multiserver models. This allows us to describe the multiqueue server as a system which contains a Zero-Server Chain that

describes when the server is not at the queue. This is shown in Figure 3.

The figure shows that the number of people in queue i increases due to the arrival rate, λ_i. The chain is exited when the server arrives at queue i again. The server will arrive at the queue again in time, T_v, which is the average of the vacation time. We define $\frac{1}{T_v}$ as the **vacation rate** represented by μ_v. This is the exit rate out the Zero-Server Chain.

A. A Markov Model for an Exhaustive Polling System

In order to make use of the Zero-Server Markov Chain in a multiserver model, we need to tie it to a Markov Chain Model which describes the service discipline while the server is at the queue. In this paper we show the simplest sytem which is the exhaustive polling model in which the server serves each queue until it is empty. This is shown in Figure 4. So the arrival of the server at queue i, causes a transition to from the Zero-Server Chain to Chain 1, where it begins to serve customers in that queue. Since the service is exhaustive the server remains serving at the queue until it is empty. So there is only one transition from Chain 1 to the Chain 0, which is from state $1,1$ to state $0,0$ as shown in the figure.

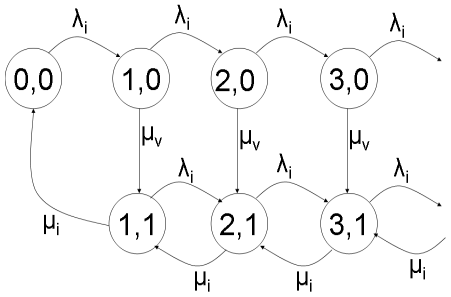

Fig. 4: Markov Chain showing Exhaustive Polling Model

B. The Analysis

To analyse this model, we represent the arrival rate at queue i, λ_i, as λ. In addition, the rate of service at queue i, μ_i is represented as μ_1 since it only operates in Chain 1 of the model.

So our first equation is:

$$\lambda p_{0,0} = \mu_1 p_{1,1} \tag{18}$$

From Chain 0:

$$p_{n,0} \quad = (\tfrac{\lambda}{\lambda+\mu_v})^n p_{0,0} \tag{19}$$

So we can use this to solve equations for Chain 1:

$$(\lambda+\mu_1)p_{1,1} \quad = \mu_v p_{1,0} + \mu_1 p_{2,1} \tag{20}$$

But from Equation 19,

$$(\lambda+\mu_v)p_{1,0} \quad = \lambda p_{0,0} \tag{21}$$

And from Equation 18:

$$p_{1,0} \quad = \tfrac{\lambda}{\mu_1} p_{0,0} \tag{22}$$

Substituting and re-arranging we get:

$$p_{2,1} \quad = p_{0,0}(\tfrac{\lambda}{\mu_1})^2(1+(\tfrac{\mu_1}{\lambda+\mu_v})) \tag{23}$$

Thus we use the equation for $p_{n-1,1}$ to solve for $p_{n,1}$. Hence:

$$(\lambda+\mu_1)p_{2,1} \quad = \lambda p_{1,1} + \mu_v p_{2,0} + \mu_1 p_{3,1} \tag{24}$$

It can be shown that:

$$p_{n,1} \quad = p_{0,0}(\tfrac{\lambda}{\mu_1})^n(1+(\tfrac{\mu_1}{\lambda+\mu_v})+...+(\tfrac{\mu_1}{\lambda+\mu_v})^{n-1}) \tag{25}$$

We can express this as:

$$p_{n,1} \quad = p_{0,0}(\tfrac{\lambda}{\mu_1})^n \sum_{m=0}^{n-1}(\frac{\mu_1}{\lambda+\mu_v})^m \tag{26}$$

In order to get a value of $p_{0,0}$, we need to sum the two chains. So:

$$\sum_{n=0}^{\infty} p_{n,0} + \sum_{n=1}^{\infty} p_{n,1} = 1 \tag{27}$$

$$\begin{aligned}\sum_{n=0}^{\infty} p_{n,0} &= \sum_{n=0}^{\infty}(\frac{\lambda}{\lambda+\mu_v})^n p_{0,0} \\ &= \frac{\lambda+\mu_v}{\mu_v} p_{0,0}\end{aligned} \tag{28}$$

The sum of the probabilities for Chain 1, can be represented using Equation 26 as:

$$\sum_{n=1}^{\infty} p_{n,1} = p_{0,0} \sum_{n=1}^{\infty}(\frac{\lambda}{\mu_1})^n \sum_{m=0}^{n-1}(\frac{\mu_1}{\lambda+\mu_v})^m \tag{29}$$

In order to solve this we transpose Equation 29; hence we sum vertically instead of horizonatlly:

$$\sum_{n=1}^{\infty}(\frac{\lambda}{\mu_1})^n \sum_{m=0}^{n-1}(\frac{\mu_1}{\lambda+\mu_v})^m = \sum_{k=0}^{\infty}(\frac{\mu_1}{\lambda+\mu_v})^k \sum_{n=k+1}^{\infty}(\frac{\lambda}{\mu_1})^n \tag{30}$$

$$= \sum_{k=0}^{\infty}(\frac{\lambda}{\lambda+\mu_v})^k \frac{\lambda}{\mu_1} \sum_{n=k+1}^{\infty}(\frac{\lambda}{\mu_1})^{n-(k+1)} \tag{31}$$

$$\sum_{n=1}^{\infty} p_{n,1} = (\frac{\lambda}{\mu_1-\lambda})(\frac{\lambda+\mu_v}{\mu_v})p_{0,0} \tag{32}$$

$$\sum_{n=0}^{\infty} p_{n,0} + \sum_{n=1}^{\infty} p_{n,1} = (\frac{\mu_1}{\mu_1 - \lambda})(\frac{\lambda + \mu_v}{\mu_v})p_{0,0} = 1 \quad (33)$$

Hence

$$p_{0,0} = (\frac{\mu_1 - \lambda}{\mu_1})(\frac{\mu_v}{\lambda + \mu_v}) \quad (34)$$

Given $p_{0,0}$ we can now calculate the probability of any state in the system

C. Solving L_{EH}, the average number of customers in the system

We can represent L_{EH} as:

$$L_{EH} = \sum_{n=1}^{\infty} np_{n,0} + \sum_{n=1}^{\infty} np_{n,1} \quad (35)$$

$$\sum_{n=1}^{\infty} np_{n,0} = p_{0,0} \sum_{n=1}^{\infty} n(\frac{\lambda}{\lambda + \mu_v})^n \quad (36)$$

Let $q = \frac{\lambda}{\lambda + \mu_v}$ and substituting:

$$\sum_{n=1}^{\infty} nq^n p_{0,0} \quad = \quad \frac{q}{(1-q)^2} p_{0,0} \quad (37)$$

The second term, $\sum_{n=1}^{\infty} np_{n,1}$ can be expressed as:

$$\sum_{k=0}^{\infty} (\frac{\mu_1}{\lambda + \mu_v})^k \sum_{n=k+1}^{\infty} n(\frac{\lambda}{\mu_1})^n p_{0,0} \quad (38)$$

$$= \quad \sum_{k=0}^{\infty} (\frac{\mu_1}{\lambda + \mu_v})^k (\frac{\lambda}{\mu_1})^{k+1} \sum_{n=k+1}^{\infty} n(\frac{\lambda}{\mu_1})^{n-(k+1)} p_{0,0} \quad (39)$$

We note that from Equation 12 we have:

$$\sum_{n=m}^{\infty} n(\frac{\lambda}{(\lambda + \mu_m)})^{n-m} = \frac{m - (m-1) * r_m}{(1 - r_m)^2} \quad (40)$$

So if we let $m = k + 1$ and $r_m = \frac{\lambda}{\mu_1}$, we get:

$$\sum_{n=k+1}^{\infty} n(\frac{\lambda}{\mu_1})^{n-(k+1)} = \frac{(k+1) - kr_m}{(1 - r_m)^2} \quad (41)$$

$$= \frac{r_m}{(1 - r_m)^2} \sum_{k=0}^{\infty} (\frac{\lambda}{\lambda + \mu_v})^k ((k+1) - kr_m) \quad (42)$$

Remembering that $q = \frac{\lambda}{\lambda + \mu_v}$ the above expression becomes:

$$\frac{r_m}{(1 - r_m)^2} \sum_{k=0}^{\infty} q^k ((k+1) - kr_m) \quad (43)$$

Let $m = k + 1$:

$$\sum_{k=0}^{\infty} q^k (k+1) = \sum_{m=1}^{\infty} mq^{m-1} = \frac{1}{(1-q)^2} \quad (44)$$

$$r_m \sum_{k=0}^{\infty} kq^k = r_m q \sum_{k=1}^{\infty} kq^{k-1} = \frac{r_m q}{(1-q)^2} \quad (45)$$

$$\frac{r_m}{(1 - r_m)^2} \sum_{k=0}^{\infty} q^k ((k+1) - kr_m) = \frac{r_m (1 - r_m q)}{(1 - r_m)^2 (1 - q)^2} \quad (46)$$

So we can write L_{EH}:

$$L_{EH} = p_{0,0} (\frac{q}{(1-q)^2} + \frac{r_m (1 - r_m q)}{(1 - r_m)^2 (1 - q)^2}) \quad (47)$$

$$= \quad p_{0,0} (\frac{q}{(1-q)^2 (1 - r_m)} + \frac{r_m}{(1-q)(1 - r_m)^2}) \quad (48)$$

Remembering that $r_m = \frac{\lambda}{\mu_1} = \rho_1$ and $q = \frac{\lambda}{\lambda + \mu_v}$:
The result is:

$$p_{0,0} (\frac{\lambda + \mu_v}{\mu_v})(\frac{\lambda}{\mu_v (1 - \rho_1)} + \frac{\rho_1}{(1 - \rho_1)^2}) \quad (49)$$

The final result can be written as:

$$L_{EH} = \frac{\lambda}{\mu_v} + \frac{\rho_1}{1 - \rho_1} \quad (50)$$

Since $T_v = \frac{1}{\mu_v}$: this is the average vacation time:
So we get:

$$L_{EH} = \lambda T_v + \frac{\rho_1}{1 - \rho_1} \quad (51)$$

Remembering that exhaustive service means that the queue is always empty when the server leaves the queue, all the mathematics yields an intuitive result: the average number of people in the queue for exhaustive service is given by the average number of customers that arrive during the vacation period, T_v plus the average number of customers that are served when the server arrives at the queue. The latter value is the average number of people that are served in an M/M/1 queue. This makes sense because if $T_v = 0$ which means that the server never leaves the queue, we get the same results for an M/M/1 queue.

VI. An Approximate solution for Symmetrical Multiqueue Systems

Let us suppose that we have n identical queues being served in an exhaustive way by a single server, we need to calculate the vacation time, T_v. We look at the time the server takes to arrive back at the queue. So when it leaves queue i, it will use up C_0 time to switchover to all the queues in one cycle. We assume that C_0 is constant. At each queue, the average number of customers is the queue is also given by Equation 51. The server must serve customers in $(n-1)$ queues with mean time $\frac{1}{\mu_i}$ which is represented as $\frac{1}{\mu_1}$. The results are as follows:

$$T_v = C_0 + (n-1)(\lambda T_v + \frac{\rho_1}{1-\rho_1})\frac{1}{\mu_1} \tag{52}$$

$$T_v = \frac{C_0}{(1-(n-1)\rho_1)} + \frac{(n-1)\rho_1}{\mu_1(1-\rho_1)(1-(n-1)\rho_1)} \tag{53}$$

$$L_{EH} = \frac{\lambda C_0(1-\rho_1)+\rho_1}{(1-\rho_1)(1-(n-1)\rho_1)} \tag{54}$$

$$W_{EH} = \frac{C_0}{(1-(n-1)\rho_1)} + \frac{\rho_1}{\lambda(1-\rho_1)(1-(n-1)\rho_1)} \tag{55}$$

Again, if $n = 1$ and $C_0 = 0$, we get the result for an M/M/1 queue.

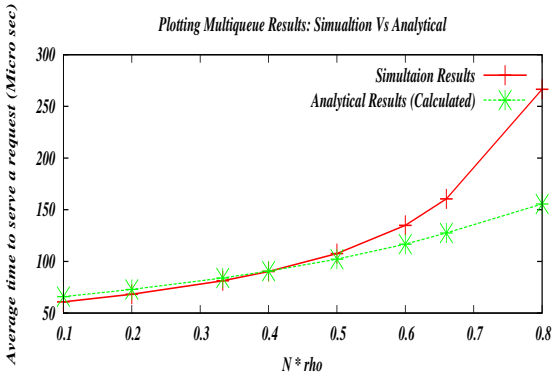

Fig. 5: Preliminary Results for N=2

A. Preliminary Results

In Figure 5, some preliminary results of the approximate model are presented where the number of queues, $N = 2$. As expected, the model captures the waiting times well until $N\rho = 0.6$. After this the waiting times of the simulation increase significantly. This is because the approximate model assumes that there is no correlation between consecutive values of T_v. This is not true as T_v for one queue is influenced by the number of people served in the previous cycle in another queue, especially when the system is heavily loaded. A more accurate model to calculate T_v is discussed in [14] but the approximate result presented above can be used as a back-of-the-envelope formula for systems using exhaustive service under normal operational loads.

VII. Conclusions and Future Work

This paper introduced a new Markov Chain model which can be used to model both multiserver and multiqueue systems. This was done by introducing the idea of a Zero-Server Markov Chain to represent multiqueue polling systems. The model was used to calculate the average number of customers in a multiqueue system with exhaustive service. An intuitive result was demonstrated by this model and an approximate solution for the waiting time for a symmetrical multiqueue system was described. We are developing algorithms to calulate more accurate values of T_v. In addition, we are seeking to model other multiqueue service models, such as non-exhaustive [18] service using this technique.

References

[1] C. Mack, "The efficiency of N machines uni-directionally patrolled by one operative when walking time is constant and repair times are variable ," *Journal of the Royal Statistical Society Series B*, vol. 19, no. 1, pp. 173–178, 1957.

[2] W. Bux, "Performance issues in local-area networks," *IBM Systems Journal*, vol. 23, no. 4, pp. 351–374, 1984.

[3] M. Kyrm, *Performance Analysis of A Local Area Network with Application to Fiber Optic Integrated Voice and Data*, Carleton University, August 1984, Master's Thesis.

[4] J. Sykes, "Simplified analysis of alternating-priority queueing model with setup times," *Operations Research*, vol. 18, no. 6, pp. 1182–1192, Nov-Dec 1970.

[5] Y. Aminetzah, *An exact approach to the polling System (PhD Thesis)*, Dept of Electrical Engineering, McGill Univ., Montreal, Quebec, 1975.

[6] R. B. Cooper, *Introduction to Queueing Theory, 2d ed.* Elsevier North-Holland, New Yoork, 1981.

[7] O. Gemikonakli, T. Do, R. Chakka, and E. Ever, "Numerical Solution to the Performability of a Multiprocessor System with Reconfiguration and Rebooting Delays," in *Proceedings of ECMS 2005*, June 2005, pp. 766–773.

[8] Q. Zeng, K. Mukumoto, and A. Fukuda, "Performance analysis of mobile cellular radio system with priority reservation and handoff procedures," in *Proceedings of IEEE VTC 1994*, June 1994, pp. 1829–1833.

[9] K. Trivedi and X. Ma, "Performability Analysis of wireless cellular networks," in *Proceedings of Symposium on Performance Evaluation of Computer and Telecommunication System (SPECTS 2002)*, San Diego, USA, July 2002.

[10] H. Takagi, *Queuing analysis of polling models*. ACM, 1988.

[11] K. Leung, "Cyclic-Service Systems with Probabilistically-Limited Service," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 2, February 1991.

[12] K. Chang and D. Sandhu, "Pseudo-conservation laws in cyclic-service systems with a class of limited service policies," *Annals of Operations Research*, vol. 35, no. 3, June 1992.

[13] R. Dittmann and F. Hubner, "Discrete-Time Analysis of a Cyclic Service System with Gated Limited Service," Institute of Computer Science, University of Wurzburg, Tech. Rep., June 1993.

[14] M. van Vuuren and E. Winands, "Iteractive approximation of k-limited polling systems," Technische Universiteit Eindhoven, Tech. Rep., May 2006.

[15] I. MacPhee, M. Menshikov, D. Petritis, and S. Popov, "A Markov Chain Model of a Polling System with Paramter Regeneration," *The Annals of Applied Probability*, vol. 17, no. 5/6, pp. 1447–1473, 2007.

[16] D. Gross and C. M. Harris, *Fundamentals of Queueing Theory (Wiley Series in Probability and Statistics)*. Wiley-Interscience, February 1998.

[17] G. Mapp, D. Thakker, and . Gemikonakli, "Exploring Gate-Limited Analytical Models for High Performance Network Storage Servers," in *Proceedings of the 3rd International Workshop on Performance Modeling and Evaluation in Computer and Telecommunication Networks (PMECT 2009)*, San Francisco, USA, August 2009.

[18] P. Keuhn, "Multiqueue Systems with Non-Exhaustive Cyclic Service," *Bell System Technical Journal*, vol. 58, pp. 671–699, March 1979.