Journal of the Nigerian Association of Mathematical Physics
Volume 46 (**May,** *2018 Issue*), **pp359 – 364**

PERFORMANCE OF A THREE-QUEUE POLLING SYSTEM WITH PROBABILISTIC ROUTING

Nehemiah Ikoba[1] and Titi Obilade[2]

[1]Department of Statistics, University of Ilorin, Ilorin, Nigeria
[2]Department of Mathematics, Obafemi Awolowo University, Ile-Ife, Nigeria

Abstract

This research is aimed at modelling a specialized three-queue priority polling system with zero switchover times. The server dynamically allocates the system resources between two queues after serving the first queue. At near saturation levels, it is necessary to determine the performance of this routing discipline. With the assumption of poisson arrivals and exponential or Erlangian service times, the system equations were obtained via the embedded Markov chain approach. The waiting time distribution was also derived and compared with results from an appropriate simulation model. Erlangian distributed service times produced lower waiting times when compared with exponentially distributed service times for the same stream of input parameters. Hence the Erlang distribution provides a better alternative in modelling service times than the exponential distribution. This priority polling system could be deployed as a solution to certain production systems where three classes of products are produced. The polling model is also proposed as an appropriate fit for traffic systems with greedy server routing. The closed form solutions may prove to be extremely useful for system design and optimization in application areas as diverse as telecommunications, manufacturing, logistics, transportation and maintenance.

Keywords: Polling model, polling probability, waiting times, probability generating function, Laplace-Stieltjes transform, super cycle.

1.0 Introduction

The use of the Asynchronous Transfer Mode (ATM) switching technology and Internet Protocol (IP) for telecommunications applications is a key aspect of the industry. At near saturation levels of such systems, when IP networks become heavily loaded, packets get dropped and several other key performance measures like packet latency, loss and jitter are significantly downgraded, leading to loss of communication in some instances. The need to assign higher priority to some class of traffic becomes pertinent, particularly as there is no dedicated communication path for any of the traffic classes (e.g voice, video, data).

Priority queueing is useful for making sure that mission-critical traffic in telecommunications applications gets priority treatment [1].

The probabilistic routing discipline is frequently being used within the internet to address quality-of-service (QoS) problems for multiple traffic classes. Under this discipline, the bandwidth is shared in accordance with the priority and polling probability assigned to the different classes of traffic. It assigns weights to classes of traffic rather than individual flows of traffic identified by origination/destination pairs. However, there is an absence of the actual analysis of the system largely due to tractability issues.

Corresponding Author**:** Nehemiah I., Email: ikoba.na@unilorin.edu.ng, Tel: +2347039466736, +2348034059544 (TO)

A direct simulation of the system is usually employed to evaluate the performance of such polling systems. Another approach is to develop the system equations which can then be solved numerically. In this paper, we use both the analytical approach and simulation, with greater emphasis on simulation in the development of the key performance analysis tools for the model.
In section 2, we present a description of the model and develop the steady-state system equations for the state probabilities. The simulation model is presented in section 3, while some results of the simulation experiment are presented and discussed in section 4 in order to highlight the performance of the system. Finally, our conclusions are expressed in section 5.

2.0 System Description

The polling system consists of three classes of packets corresponding to three different queues. This may be the case at a traffic junction consisting of one traffic controller passing cars from three different routes.
We assume independent Poissonian arrivals with rates $\lambda_1, \lambda_2, \lambda_3$ for each of the queues at the junction, and the distribution of the service times follow a general distribution. For illustrative purposes, the exponential and Erlang distributions were used in modelling the service times for the simulation exercise.
The system is non-preemptive. Upon service completion of a packet, the next type of packet to be served is randomly chosen according to the polling or selection probability p.
The service times at queue i are independent and identically distributed random variables B_i with finite k^{th} moments $b_i^{(k)}$, k=1,2,...
When the server, working in exhaustive fashion, has finished with the batch of services at queue i or if it finds that queue empty, it switches to the next queue without incurring a switchover time. The choice of the queue may depend on the current state of the system through the polling probability.
A necessary and sufficient condition for stability is that the overall system load or server utilization ρ, is less than 1.
There is a large body of research on polling models. We mention a few papers that specifically deal with polling models that are related to this model with probabilistic routing. A related polling system was analyzed in [2], where the polling order was periodic, with exhaustive discipline at each queue. The analysis involved the study of the embedded process at four points namely: service beginning, service completion, visit beginning and visit completion. A similar work by [3] provided closed and exact expressions for some key performance measures of the system in a two-queue model.
A processor-sharing approximation was derived by [4] for the strictly alternating switch (SAS) polling model. The SAS is basically an endogenous priority queueing model in which the next customer for service is selected not only on what priority class it belongs but also on what priority class was last served. It represents a particular extreme case of switching from one queue to the other after a specified k≥ 1 number of units have been served in a two-queue system. The two queues are M/M/1-type and a feedback model was integrated into the system, in which either unit may require further service with some probability.
A probabilistic polling system with Bernoulli scheduling order of service was studied in [5]. Another probabilistic case mentioned by [5] is one in which, after the completion of service at any queue, the next polled queue is queue j with probability p_j, where $\sum_N^{j=1} p_j = 1$, called random polling.
A discrete-time random polling model was considered in [6] under different service disciplines. For the limited-service discipline, results were provided only for a completely symmetric system. For the general case of N queues with the exhaustive service discipline, a set of N^2 linear equations were derived in [6], and the equations could then be solved to obtain the expected waiting times. Exact results were provided in [7] for the asymmetric case of a random polling model, while a simulation framework was provided in [8] for studying the performance of a class-based weighted fair queueing routing discipline, using a custom-made simulator.
A unique property of the present model is that the server's routing rule depends on the actual configuration of customers in the system. The embedded Markov chain technique was used to obtain the steady state system equations. We are concerned at any instant t, with a group of random variables N(t), the number of customers in the system at time t, and X(t) the service time already received by the customer in service, if any. {N(t), t≥0} is non-Markovian, but {N(t),X(t), t≥0} is a Markov process.
Now, by observing the number in the system at the switch points - the instant the server exhausts a particular queue and is about to switch to the next, rather than at all points in time t, it is possible to simplify matters to a great extent.
Let $\tau_1, \tau_2, \dots, \tau_N$ be the time instants at the switch points. At the switch point $X(\tau)=0$, since the last customer just completed service, thus effectively reducing the dimension of our embedded Markov Chain to $\{N(\tau_i), 0, \tau_i \geq 0\}$.
Define $P_i(q_{i-1}, 0, q_{i+1}) = \Pr\{N_{i-1}(\tau) = q_{i-1}, N_i(\tau) = q_{i+1}\}$ as the joint probability that at an arbitrary switch point, the server has just completed a visit to queue i and q_j customers are waiting in queue j, (j=i-1,i+1), i=1,2,3.
The possible server cycles according to the model are:
(1) 1-2-3 (Cycle 1, C_1), $\Pr(C_1)$=1-p; or (2) 1-3 (Cycle 2, C_2), $\Pr(C_2)$=p.
Drawing from the analysis of the model in [9], the probability state equations for queue i=1,2 are obtained as

$$\begin{aligned} P_i(q_{i-1}, 0, q_{i+1}) = &\sum_{k_i=1}^{\infty} \sum_{k_{i+1}=0}^{q_{i+1}} P_i(0, k_i, k_{i+1}) \int_0^{\infty} \frac{(\lambda_{i+1} t)^{q_{i+1}-k_{i+1}}}{(q_{i+1}-k_{i+1})!} e^{-\lambda_{i+1} t} \frac{(\lambda_{i-1} t)^{q_{i-1}}}{(q_{i-1})!} e^{-\lambda_{i-1} t} d\theta_i^{(k_i)}(t) \\ &+ P_i(0,0,q_{i+1})\big(1-\delta(q_{i+1})\big)\delta(q_{i-1}) \\ &+ \frac{\lambda_i}{\lambda} P(0) \int_0^{\infty} \frac{(\lambda_{i+1} t)^{q_{i+1}}}{(q_{i+1})!} e^{-\lambda_{i+1} t} \frac{(\lambda_{i-1} t)^{q_{i-1}}}{(q_{i-1})!} e^{-\lambda_{i-1} t} d\theta_i(t) \end{aligned} \quad (2.1)$$

Where $P(0) = \sum_{i=1}^{3} P_i(0,0,0)$

$$\delta(x) = \begin{cases} 1, & if\ x = 0 \\ 0, & if\ x \neq 0 \end{cases}$$

and $\theta_i(t)$ is the busy period distribution function while $\theta_i^{(k_i)}$(t) is its k_i-fold convolution.

The normalization condition is given by

$$\sum_{i=1}^{3} \sum_{q_{i-1}=0}^{\infty} \sum_{q_{i+1}=0}^{\infty} P_i(q_{i-1}, 0, q_{i+1}) = 1$$

In order to compute $P_3(q_1, q_2, 0)$, we have to condition it on the server cycle so as to correctly specify the system at a 3-switch point. The conditional probabilities $P_3(q_1, q_2, 0|C1)$ and $P_3(q_1, q_2, 0|C2)$ are first derived:

$$P_3(q_1, q_2, 0|C1) = \sum_{k_3=1}^{\infty} \sum_{k_1=0}^{q_1} P_2(k_1, 0, k_3) \int_0^{\infty} \frac{(\lambda_1 t)^{q_1-k_1}}{(q_1-k_1)!} e^{-\lambda_1 t} \frac{(\lambda_2 t)^{q_2}}{(q_2)!} e^{-\lambda_2 t} d\theta_3^{(k_3)}(t) + P_3(0,0,q_1)\left(1-\delta(q_1)\right)\delta(q_2) + \frac{\lambda_3}{\lambda} P(0) \int_0^{\infty} \frac{(\lambda_1 t)^{q_1}}{(q_1)!} e^{-\lambda_1 t} \frac{(\lambda_2 t)^{q_2}}{(q_2)!} e^{-\lambda_2 t} d\theta_3(t) \quad (2.2)$$

and

$$P_3(q_1, q_2, 0|C2) = \sum_{k_3=1}^{\infty} \sum_{k_2=0}^{\infty} P_1(0, k_2, k_3) \int_0^{\infty} \frac{(\lambda_1 t)^{q_1}}{(q_1)!} e^{-\lambda_1 t} \frac{(\lambda_2 t)^{q_2-k_2}}{(q_2-k_2)!} e^{-\lambda_2 t} d\theta_3^{(k_3)}(t) + P_1(0,q_2,0)\left(1-\delta(q_2)\right)\delta(q_1) + \frac{\lambda_3}{\lambda} P(0) \int_0^{\infty} \frac{(\lambda_1 t)^{q_1}}{(q_1)!} e^{-\lambda_1 t} \frac{(\lambda_2 t)^{q_2}}{(q_2)!} e^{-\lambda_2 t} d\theta_3(t) \quad (2.3)$$

The probability, $P_3(q_1, q_2, 0)$ is the weighted average of both $P_3(q_1, q_2, 0|C1)$ and $P_3(q_1, q_2, 0|C2)$. That is

$$P_3(q_1, q_2, 0) = P_3(q_1, q_2, 0|C1)\Pr(C1) + P_3(q_1, q_2, 0|C2)\Pr(C2) \quad (2.4)$$

The joint probability generating function for i=1,2 is

$$G_i(z_{i-1}, 0, z_{i+1}) = G_{i-1}\left(0, \theta_i^*\left[\textstyle\sum_{j\neq i} \lambda_j\left(1-z_j\right)\right], z_{i+1}\right) - P_{i-1}(0,0,0)\frac{\lambda_i}{\lambda} P(0)\theta_i^*\left[\textstyle\sum_{j\neq i}\lambda_j\left(1-z_j\right)\right] \quad (2.5)$$

Where $\theta_i^*(s)$ is the Laplace-Stieltjes transform (LST) of the type-i busy period distribution function, and is given by

$$\theta_i^*(s) = B_i^*(s + \lambda_i - \lambda_i\theta_i^*(s))$$

The joint probability generating function for i=3 is given by

$$G_3(z_1, z_2, 0) = (1-p)G_2\left(z_1, 0, \theta_3^*\left[\sum_{j\neq 3}\lambda_j\left(1-z_j\right)\right]\right) + pG_1\left(0, z_2, \theta_3^*\left[\sum_{j\neq 3}\lambda_j\left(1-z_j\right)\right]\right) - pP_1(0,0,0) - (1-p)P_2(0,0,0) + \frac{\lambda_3}{\lambda}P(0)\theta_3^*\left[\sum_{j\neq 3}\lambda_j\left(1-z_j\right)\right] \quad (2.6)$$

By differentiating equations (2.5) and (2.6) with respect to z_j and z_k and then setting z_j=$z_k = 1$ we have a set of $3^2 = 9$ recursive equations for $\{g_i(j,k)$; i,j,k=1,2,3, i≠j,k}. The expression $g_i(j,k)$ is the cross correlation of the mean queue lengths at queues j and k at an i-switch point. The values of these cross correlations are obtained by solving the set of 9 equations obtained from equations (2.5) and (2.6). Such recursive equations are amenable to numerical solutions, as results would converge numerically in a reasonable number of steps.

In providing an analytical solution to the system, we define a *super cycle* as the elapsed time between the arrival instant of a customer at any queue when the system is empty, and the first instant at which the system becomes empty again.

We also classify the customers that arrive into queue i into two exclusive and exhaustive types:

1) arrivals that either initiate a super cycle or occur during the first busy period in a super cycle; or
2) all other arrivals that occur after (and including) the second busy period at queue i in a super cycle.

The LST of the waiting times distribution at queue i=1,2, is given as

$$W_i^*(s) = \frac{s(1-\rho)}{s-\lambda_i+\lambda_i B_i^*(s)} + \frac{\lambda_i(\rho-\rho_i)(G_{i-1}(0,1,1) - G_{i-1}\left(0, 1-{}^{s}/_{\lambda_i}, 1\right))}{g_{i-1}(i)(s-\lambda_i+\lambda_i B_i^*(s))} \quad (2.7)$$

The LST of the waiting times distribution for queue 3, $W_3^*(s)$ is

$$W_3^*(s) = \frac{s(1-\rho)}{s-\lambda_3+\lambda_3 B_3^*(s)} + \frac{\left[(\rho-\rho_3)\left\{G_1(0,1,1)\left[G_2(1,0,1) - (1-p)G_2\left(1,0,1-{}^{s}/_{\lambda_3}\right)\right] - G_2(1,0,1)pG_1\left(0,1,1-{}^{s}/_{\lambda_3}\right)\right\}\right]}{[(1-p)G_1(0,1,1)g_2(3) + pG_2(1,0,1)g_1(3)](s-\lambda_3+\lambda_3 B_3^*(s))} \quad (2.8)$$

The expected waiting time at queue i, E(W_i) for i=1,2, is given as

$$E(W_i) = -\frac{\partial}{\partial s} W_i^*(s)|_{s=0} = \frac{\lambda_i b_i^{(2)}}{2(1-\rho_i)} + \frac{g_{i-1}(i,i)}{2\lambda_i^2(1-\rho_i)} \quad (2.9)$$

and

$$E(W_3) = \frac{\lambda_3 b_3^{(2)}}{2(1-\rho_3)} + \frac{(1-p)\lambda_1 g_2(3,3) + p\lambda_2 g_1(3,3)}{2\lambda_3^2(1-\rho_3)((1-p)\lambda_1 + p\lambda_2)} \quad (2.10)$$

Equation (2.10) thus establishes the mean waiting time at queue 3 and its relationship with the second factorial moments of the marginal queue length distribution in queue 3 when the server switches either from queue 1 ($g_1(3,3)$) or queue 2 ($g_2(3,3)$). Hence $E(W_3)$ is a function of the amount of dispersion in the marginal queue length at queue 3, and it is inversely proportional to the polling probability, p.

It is interesting to note that for the case p=0, $E(W_3)$ reduces to

$$E(W_3) = \frac{\lambda_3 b_3^{(2)}}{2(1-\rho_3)} + \frac{g_2(3,3)}{2\lambda_3^2(1-\rho_3)} \quad (2.11)$$

The above expression, as derived in [9], coincides with previously obtained results by [2] and [10].

3.0 Simulation Model

The development of a customized simulation model was motivated by the inadequacies of existing simulation packages like MATLAB, Maple, and OPNET modeler, to effectively capture the polling model with probabilistic routing. These simulators have limited ability to model router configurations inherent in the model with probabilistic routing. We desired a flexible simulator that could be modified to examine future system configurations and effectively capture router configuration limitations.

The simulation model was built using blocks from SIMULINK, a general simulation software which uses both programmed and programmable blocks in performing the simulation. The integration of SIMULINK with MATLAB provides immediate access to an extensive range of tools that allows one to develop algorithms, analyze and visualize simulations, customize the modeling environment, and define a signal parameter and test data. It enables one to pose a question about a system, model the system and observe the evolution of the system. One of the most important blocks in the simulation model is the path combiner, which chooses the queue to be accessed by the server according to the state of the system. Much work in formulating the simulation model was involved in configuring the path combiner to fit the model specifications, using compound logic. The simulation model is as shown in figure 1.

The simulator computes several measures of effectiveness: waiting times or latency, mean system size, server utilization, number of packets served, etc. It can also output individual queue waiting times and number of packets at each service completion instants, making it possible to obtain the entire distributions if needed.

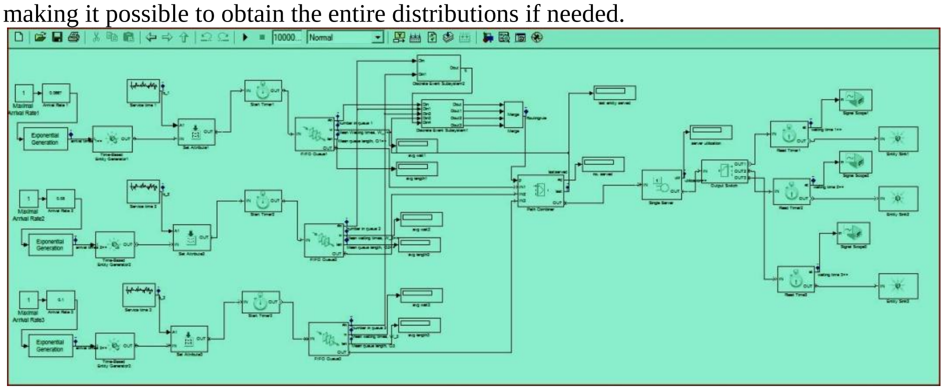

Figure 1: The SIMULINK simulation model

4.0 Results and Discussions

The input parameters used in the simulation exercise are as given in table 1. About 360 simulations were carried out using various combinations of the input parameters and polling probability p. A comparison of analytical results with the corresponding simulation results and the accompanying 95% confidence interval for the mean waiting times, resulting from exponentially distributed and Erlang distributed service times are presented in tables 2 and 3 respectively.

Customers at a heavily loaded queue experienced lower delays than customers at lightly loaded queues. This is because the server is more likely to be at the heavily loaded queue when a customer arrives. To a lesser extent, the lightly loaded queue that is close to the heavily loaded queue in the direction the server is moving also enjoys lower delays.

When all queues have almost the same load levels, the waiting times for p=0 are similar. This validates the result that for exhaustive service disciplines, symmetric queues have similar system parameters (waiting times, queue lengths, etc), that is, they are statistically identical.

Increasing levels of p for a lightly loaded system ($\rho \leq 0.5$) do not produce a drastic change in the waiting times. This is because the server is never idle while there is work waiting (i.e it is a work-conserving system).

This means that after following the specified polling order, if the server still finds only queue 2 non-empty at any switch point, it goes straightaway to queue 2 and serves it in an exhaustive fashion. This tends to stabilize the waiting times across queues, as the proportion of time the server is busy is not too high. On the other hand, increasing levels of p for a heavily loaded system produces sometimes dramatic changes in the waiting times. Sharp increases are especially found in queue 2.
Because of the exhaustive nature of the service process, coupled with the fact that the proportion of the server's resources spent on any queue is small, the buffer size at each queue does not affect the waiting times in the system at low load levels. However, at near saturation levels, the buffer size becomes an important factor that affects the waiting times. There is a steady rise in the waiting times as the buffer sizes increase, until a point is reached when the server is put to full use and waiting times start reducing with increasing buffer sizes, with the waiting times approaching the infinite buffer cases.
The simulation model provided a good fit for the system, as over 85% of theoretical values fell within the 95% confidence interval obtained from simulation results. In fact, 89% was observed for exponentially distributed service times while 87% was observed for Erlang distributed service times.
The simulation runs provided a means of comparison of the exponential and Erlang distributions. Erlangian distributed service times produced lower queue lengths and waiting times when compared with exponentially distributed service times for the same stream of input parameters, as could be seen from tables 2 and 3.

Table 1: Input parameters and the resulting load (ρ_i) used in the simulation exercise

b_i (minutes)	$\lambda_1 = 1/15$	$\lambda_2 = 1/20$	$\lambda_3 = 1/10$
1	ρ_1 =0.0667	ρ_2 =0.05	ρ_3 =0.1
1.5	ρ_1 =0.1	ρ_2 =0.075	ρ_3 =0.15
2	ρ_1 =0.1333	ρ_2 =0.1	ρ_3 =0.2
3	ρ_1 =0.2	ρ_2 =0.15	ρ_3 =0.3
4	ρ_1 =0.2667	ρ_2 =0.2	ρ_3 =0.4
4.5	ρ_1 =0.3	ρ_2 =0.225	ρ_3 =0.45

Table 2: Waiting times comparison of theoretical and simulation results and the 95% confidence interval for the mean waiting times (Exponential service times)

ρ_1 ρ_2 ρ_3	p	W_1 (minutes)			W_2 (minutes)			W_3 (minutes)		
		Theory	Sim	C I	Theory	Sim	C I	Theory	Sim	C I
0.133 0.1 0.45	0	10.6	10.48	9.45,11.38	11.5	11.56	10.23,13.14	6.2	6.09	5.80,6.72
	0.3	10.3	10.45	8.90,11.24	12.5	12.03	11.04,13.97	6.0	6.21	5.34,7.11
	0.5	10.1	10.60	8.43,12.59	13.4	12.61	11.86,14.54	5.4	6.03	4.96,6.82
	0.7	9.8	9.85	9.12,10.36	14.4	14.13	10.28,16.42	5.2	5.73	5.01,6.13
	0.9	9.2	9.73	8.81,10.67	15.1	14.56	12.02,17.53	5.0	5.31	4.84,6.15
0.2 0.15 0.45	0	19.4	18.59	16.87,20.31	20.3	21.33	19.49,23.17	12.4	12.27	11.38,13.16
	0.3	18.6	18.76	17.25,20.27	22.1	21.64	19.83,23.45	11.5	12.25	11.42,13.08
	0.5	16.9	18.44	16.64,20.24	26.2	24.38	21.99,26.77	10.6	11.98	10.77,13.19
	0.7	15.6	17.55	15.60,19.50	30.2	28.56	25.46,31.66	9.6	10.58	9.67,11.49
	0.9	15.0	17.00	16.37,17.63	33.1	32.49	31.08,33.90	9.0	9.74	9.51,9.97
0.267 0.2 0.4	0	26.3	27.41	25.32,29.50	31.5	30.88	28.37,33.39	21.2	22.29	20.72,23.86
	0.3	25.6	26.56	25.09,28.03	32.3	30.38	28.59,32.17	20.0	21.20	19.88,22.52
	0.5	25.4	25.87	23.90,27.84	36.1	35.20	33.11,37.29	18.9	19.87	18.78,20.96
	0.7	23.4	25.75	23.16,28.34	46.2	45.67	41.50,49.84	16.8	17.89	16.41,19.37
	0.9	22.1	22.85	21.52,24.18	51.3	45.69	42.38,49.00	14.6	15.10	14.31,15.89

Key: p – polling probability, Sim – Simulation, CI – Confidence Interval

Table 3: Waiting times comparison of theoretical and simulation results and the 95% confidence interval for the mean waiting times (Erlang service times)

ρ_1 ρ_2 ρ_3	p	W_1 (minutes)			W_2 (minutes)			W_3 (minutes)		
		Theory	Sim	C I	Theory	Sim	C I	Theory	Sim	C I
0.133 0.1 0.45	0	6.52	6.71	6.28,7.14	7.3	7.61	7.23,7.99	4.5	3.94	3.79,4.09
	0.3	6.8	6.8	6.37,7.23	7.73	7.85	7.33,8.37	4.0	3.86	3.70,4.02
	0.5	6.9	6.78	6.55,7.01	8.0	8.21	7.76,8.66	3.9	3.89	3.80,3.98
	0.7	6.3	6.72	6.24,7.20	8.9	9.25	8.78,9.72	3.7	3.69	3.54,3.84
	0.9	6.1	6.39	6.09,6.70	10.2	9.89	9.47,10.31	3.4	3.44	3.33,3.55
0.2 0.15 0.45	0	12.0	11.79	11.11,12.47	14.1	13.23	12.56,13.90	7.65	7.70	7.34,8.06
	0.3	11.21	11.28	10.67,11.89	15.2	13.04	12.22,13.86	7.01	7.26	6.94,7.58
	0.5	10.8	11.22	10.72,11.72	17.4	14.14	13.31,14.97	6.8	7.09	6.79,7.39
	0.7	10.58	10.28	10.33,11.43	19.2	17.84	16.86,18.82	6.5	6.51	6.18,6.84
	0.9	10.1	10.38	10.2105,10.71	20.1	19.40	18.67,20.13	5.8	5.97	5.79,6.15
0.267 0.2 0.4	0	17.6	16.58	15.35,17.81	20.4	18.26	16.88,19.64	14.4	13.55	12.63,14.47
	0.3	17.1	17.02	16.35,17.69	19.61	19.72	18.98,20.46	13.83	13.64	13.02,14.26
	0.5	15.4	15.53	12.78,18.28	24.2	22.38	21.02,23.74	12.0	12.67	11.91,13.43
	0.7	15.2	15.50	15.15,15.85	29.1	27.93	26.77,29.09	10.6	11.01	10.58,11.44
	0.9	14.1	14.43	13.80,15.06	32.3	29.59	28.21,30.97	9.1	9.58	9.20,9.96

Key: p – polling probability, Sim – Simulation, CI – Confidence Interval

5 Conclusion

In order to make the numerical methods tractable, we had to restrict the comparison to only three classes of packets with Poisson arrivals and exponential or Erlangian service times. These simplifying assumptions allowed us to generate the balance equations for the system.

When more realistic assumptions about the packet arrival process and packet length distributions are used, it appears that simulation will be the only tool that will yield practical results.

The Erlang family of probability distributions provides much more modeling flexibility than does the exponential. In practical situations where observed data might not bear out the exponential distribution assumption, the Erlang can provide greater flexibility by being better able to represent the real world.

It has been shown in [11] that system optimization can be achieved with this kind of probabilistic or priority servicing pattern.

The performance indices (means of queue lengths, waiting times, intervisit times and busy periods) will aid operators and practitioners in improving quality of service and reducing system idle time.

A significant advantage of the simulation model is that it outputs the actual waiting time for each customer by type, hence the distribution of the queue waits and queue lengths could easily be obtained.

We stress that simulation remains a remarkably good means of exploring the system's behaviour and its performance. This is especially true for systems whose analytical results are either not available or its use is rather complicated.

References

[1] Bejan, A. 2007. Switchover Time Modelling in Priority Queueing Systems. Ph.D Thesis, Department of Mathematics and Computer Science, State University of Moldova.

[2] Eisenberg, M. 1972. Queues with Periodic Service and Changeover Time. *Operations Research*, **20** (2), pp. 440-451.

[3] Boxma, O. J. and Down, D. G. 1997. Dynamic Server Assignment in a Two-Queue Model. *European Journal of Operational Research*, **103**, 515-609.

[4] Obilade, T. 1983. The Processor-Sharing Approximation to the Strictly Alternating Switch Model. *Scientific annals of University of Al. I Cuza, Romania*. **29** (3), pp. 35-40.

[5] Takagi, H. 1988. Queueing Analysis of Polling Models. *ACM Computing Surveys*, **20** (1).

[6] Kleinrock, L. and Levy, H. 1988. The Analysis of Random Polling Systems. *Operations Research* **36** (5), pp. 716-732.

[7] Lee, T. 1997. A Closed Form Solution for the Asymmetric Random Polling System with Correlated Levy Input Process. *Mathematics of Operations Research* **22** (2), pp. 432-457.

[8] Fischer, M. J., Masi, D. M. B., and Shortle, J. F. 2008. Simulating the Performance of a Class-Based Weighted Fair Queueing System. In *Proceedings of the 2008 Winter Simulation Conference*. S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler, Eds.

[9] Ikoba, N. A. and Obilade, T. O. 2014. Analysis of a Three-Queue Polling System with Probabilistic Routing. *Ilorin Journal of Science,* 1 (1); 87-99.

[10] Takagi, H. and Kleinrock, L. 1984. A Tutorial on the Analysis of Polling Systems. Technical Report, Defence Advanced Research Projects Agency, Japan.

[11] Levy, H. and Sidi, M. 1990. Polling Systems: Applications, Modelling, and Optimization. *IEEE Transactions on Communications*. **38** (10).