Numerical Solution to the Performability of a Multiprocessor System with Reconfiguration and Rebooting Delays

Orhan Gemikonakli
Middlesex University
White Hart Lane
London N17 8HR
United Kingdom
E-mail: o.gemikonakli@mdx.ac.uk

Tien Van Do
Dept. of Telecommunications
Budapest University of Technology
and Economics
Hungary
E-mail: do@hit.bme.hu

Ram Chakka
RGMCET
Nandyal
India 518501
E-mail: ramchakka@yahoo.com

Enver Ever
Middlesex University
White Hart Lane
London N17 8HR, UK
E-mail: e.ever@mdx.ac.uk

KEYWORDS

Queuing, multi-processor systems, rebooting and reconfiguration delays.

ABSTRACT

Multiprocessor system models are extensively used in modelling transaction processing systems, nodes in communication networks, and flexible machine shops with groups of machines. Such systems clearly, are prone to break-downs. Even if cover is provided with some probability *c*, there will be rebooting and/or reconfiguration delays to resume operation following the break-down of a processor. In this paper, the performance modelling of a multiprocessor system, with identical processors, serving a stream of arriving jobs is considered. To account for delays due to reconfiguration and rebooting, such systems are modelled and solved for *exact* performability measures for both bounded and unbounded queuing systems.

INTRODUCTION

Multiserver system models are useful to model multiprocessor systems (Trivedi 2002; Harrison and Patel 1993), nodes in communication networks, and flexible machine shops (Stecke and Kim 1989; Stecke 1992; Righter 1996; Buzacott and Shantikumar 1993; Fiems et al. 2004) in a manufacturing environment. In this paper we develop approaches to model homogeneous multiprocessor systems with reconfiguration and rebooting delays by suitably extending the resulting quasi birth death (QBD) process in the performance models of multiprocessor systems with breakdowns and repair strategies (Chakka and Mitrani 1992; Chakka et al. 2002). This was considered in (Trivedi and Sathaye 1990) and an approximate performance model based on Markov reward models was presented. In this paper, we derive an exact solution for the steady state probabilities of the same problem using the spectral expansion method. The effects of reconfiguration and rebooting delays are analysed.

Proceedings 19th European Conference on Modelling and Simulation
Yuri Merkuryev, Richard Zobel, Eugène Kerckhoffs © ECMS, 2005
ISBN 1-84233-112-4 (Set) / ISBN 1-84233-113-2 (CD)

The paper is organised as follows. The next section presents the homogeneous multiprocessor system with breakdowns and repairs considered in this work, and models the system as a QBD process. The section on modelling reconfiguration and rebooting delays in multiprocessor systems deals with a homogeneous multiprocessor system with breakdowns, repairs, and with reconfiguration and rebooting delays (Trivedi and Sathaye 1990). Exact solution for steady state performability for is derived using the spectral expansion method in the section on steady state solution. The model considered is very useful in the computer industry. Exact solution to this model and numerical results are also presented for both unbounded and bounded systems.

MULTIPROCESSOR SYSTEM WITH IDENTICAL PROCESSORS

The homogeneous multiprocessor system, shown in Figure 1, consists of K identical parallel processors, numbered 1, 2, …, K, with a common queue. The queue is of capacity L (finite or infinite $L \geq K$), including the jobs in service. Jobs arrive into the system in a Poisson stream at rate σ, and join the queue. Jobs are homogeneous and the service rates of the processors assumed identical. Thus, the service times of jobs serviced by processor k (k=1, 2, …, K) are distributed exponentially with mean $1/\mu$. However, processor k executes jobs only during its operative periods (during an operative period the processor is capable of its intended operation, whether working or idle), which are distributed exponentially with mean $1/\xi$ (equivalent to a constant failure rate of ξ when operative). At the end of an operative period, processor k breaks down and requires an exponentially distributed repair time with mean $1/\eta$. The number of repairs that may proceed in parallel could be restricted. This is expressed by saying that there are R repairmen ($R \leq K$), each of whom can work on at most one repair at a time. Thus, an inoperative period of a processor would also include the possible waiting for a repairman. No operative processor can be idle if there are jobs awaiting service, and no repairman can be idle if there are broken-down processors waiting for repair. All inter-arrival, service, reconfiguration, rebooting, operative and repair time random variables are independent of each other. The reconfiguration delay $1/\delta$ and the rebooting delay $1/\varphi$ relate to the system and not to individual processors.

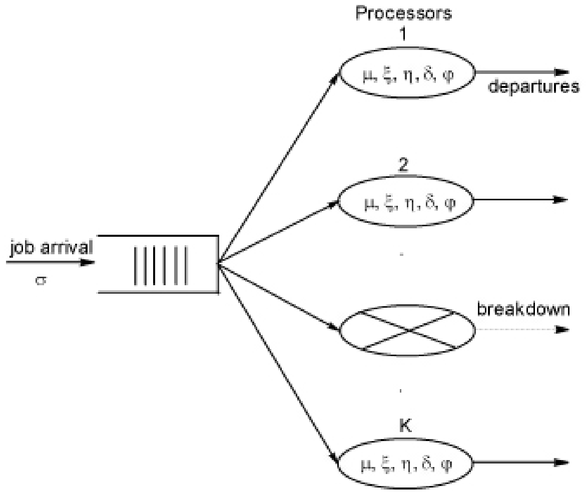

Figure 1: A Homogeneous Multiprocessor System with Breakdowns, Repairs, Reconfiguration and Rebooting Delays

If there are more operative processors than jobs in the system, then the busy processors are selected randomly. Services that are interrupted by breakdowns are eventually resumed (perhaps on a different processor but at a similar service rate). Similarly, if $R < K$ and the repair strategy allows preemptions of repairs, then interrupted repairs are eventually resumed from the point of interruption and there are no switching delays.

The state of the system at time t can be described by a pair of integer valued random variables, $I(t)$ and $J(t)$ specifying the *processor configuration* (can also be termed, *operative state* of the multiprocessor system) and *the number of jobs present*, respectively. Here, the precise meaning of processor configuration, and hence the range of values of $I(t)$, mean the number of operational processors and associated reconfiguration/rebooting delay when appropriate.

In general, let's assume that there are N+1 processor configurations, (operative states of the multiprocessor) represented by the values $I(t) = 0, 1, \dots, N$. These N+1 configurations are the *operative states* of the model. The model assumptions are assumed to ensure that $I(t)$, $t \geq 0$, is an irreducible Markov process. $J(t)$ is the total number of jobs in the system at time t, including the ones in service. Then, $Z = \{[I(t), J(t)];\ t \geq 0\}$ is an irreducible Markov process on a lattice strip (a QBD process), that models the system. Its state space is, $\{0, 1, \dots, N\} \times \{0, 1, \dots, L\}$.

This system was analysed for exact performability (Chakka and Mitrani 1994; Chakka 1995), for single repairman (R=1) and $L \rightarrow \infty$ and for some repair strategies but reconfiguration and rebooting delays were not considered.

MODELLING RECONFIGURATION AND REBOOTING DELAYS IN MULTIPROCESSOR SYSTEMS

In multiprocessor systems, in practice however, some delay is encountered when a failed processor is being mapped out of the system (reconfiguration/rebooting delay), and when a repaired processor is being admitted into the system. It is possible to model the system affected by such reconfiguration and rebooting delays effectively using the spectral expansion method.

Consider the homogeneous multiprocessor system with K processors, given in Figure 1. μ and ξ are the service and failure rates of each of the processors. There is a single repair facility (i.e. R=1) with repair rate η. When a processor fails the fault is covered with probability c and is not covered with probability 1-c. Subsequent to a covered fault, the system comes up in a degraded mode after a brief reconfiguration delay, while after an uncovered fault a longer reboot action is required to bring the system up at a degraded mode. Here, degraded mode indicates a state with one less operative processor than the previous state. For reconfiguration/rebooting period, the system is assumed to be down.

The reconfiguration and rebooting times are exponentially distributed with mean $1/\delta$ and $1/\varphi$ respectively. The queuing capacity is L, where L can be finite or infinite. σ is the arrival rate of jobs.

An approximate performance modelling of this system was carried out in (Trivedi and Sathaye 1990). We intend to carry out an exact performance evaluation of this system. Figure 2 is the Markov chain that represents the operative states of the multiprocessor system.

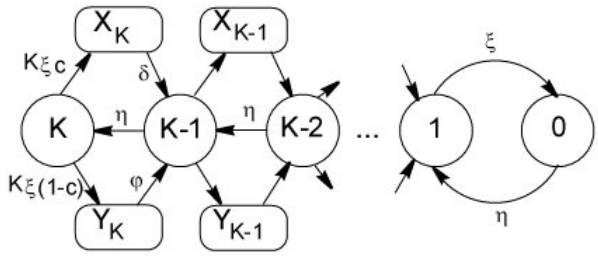

Figure 2: A Homogeneous Multiprocessor System with Breakdowns, Repairs, Rebooting and Reconfiguration Delays

The states labelled 1, 2, …, K are the K working states of the multiprocessor, with that many number of processors in each state. State 0 means no processor is operational. The K-1 states, labelled as X_2, X_3, …, X_K, are the states representing the reconfiguration delay.

The K-1 states labelled as Y_2, Y_3, …, Y_K, are the rebooting delay states. Hence, the total number of operative states is 3K-1. Let these be renumbered as, states 0, 1, …, K unchanged, states X_2, X_3, …, X_K as K+1, K+2, …, 2K-1, and the states Y_2, Y_3, …, Y_K as 2K, 2K+1, …, 3K-2.

The system now can be represented by a QBD process with finite or infinite state space. The state of the system can be defined by ($I(t)$, $J(t)$) where $I(t)$ is the operative state and $J(t)$ is the number of jobs in the system. Let the operative states be represented in the horizontal direction and the number of jobs in the vertical direction of a two-dimensional lattice strip. Here A is the matrix of instantaneous transition rates from operative state i to operative state k with zeros on the main diagonal. These are the purely lateral transitions of the model Z. Matrices B and C are transition matrices for one-step upward and one-step downward transitions respectively. When the transition rate matrices depend on j for $j \geq M$, where M is a threshold having an integer value, the process Z evolves with the following instantaneous transitions:

A_j: Purely lateral transition rate, from state (i, j) to state (k, j), ($0 \leq (i \,\&\, k) \leq N$; $i \neq k$; j=0,1, …L), caused by a change in the operative state (i.e. a break-down followed by reconfiguration or rebooting, and a repair).

B_j: One-step upward transition rate, from state (i, j) to state (k, j+1), ($0 \leq (i \,\&\, k) \leq N$; j=0,1,…L), caused by a job arrival into the queue.

C_j: One-step downward transition rate, from state (i, j) to state (k, j-1), ($0 \leq (i \,\&\, k) \leq N$; j=1,2, …L), caused by the departure of a serviced job.

Clearly, the elements of A depend on the parameters K, ξ, η, c, δ and φ. The state transition matrices A, A_j, B, B_j, C, and C_j, can be given as follows;

$$A = \begin{bmatrix} 0 & \eta & 0 & 0 & 0 & 0 & 0 & 0 \\ \xi & 0 & \eta & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \eta & 2\xi c & 0 & 2\xi(1-c) & 0 \\ 0 & 0 & 0 & 0 & 0 & 3\xi c & 0 & 3\xi(1-c) \\ 0 & \delta & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \delta & 0 & 0 & 0 & 0 & 0 \\ 0 & \varphi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \varphi & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (1)$$

In Equation (1) K=3 is assumed. Since time-dependent failures are considered, the matrices A_j do not depend on j, and hence, $A_j = A$ for all values of j. Similarly,

$B_j = B$, j=0, 1, …L;
$B = Diag$[σ, σ, …, σ] of size (3K-1)x(3K-1).

$C_j = C$ for $j \geq K$;
$C = Diag$[w(0)μ, w(1)μ, …, w(3K-2)μ];

C_0 = [**0**];
$C_j = Diag$[$Min\{w(0), j\}$μ, $Min\{w(1), j\}$μ, …,
$Min\{w(3K\text{-}2), j\}$μ] for $1 \leq j < K$

where $w(i)$ is the number of working processors in the operative state i.

We define the matrices $Q(\lambda)$ and $\overline{Q}(\beta)$ as before (Chakka 1998). Then, the steady state probabilities, $p_{i,j}$, can again be expressed in a similar manner as shown in (Chakka 1998). From this, the required performability measures such as the steady state probabilities, average number of jobs in the system, utilization of the processors, and mean response time can be obtained exactly following the computational procedure found in (Chakka 1998; Chakka 1995). Using the steady state probabilities, the response time distribution can also be derived.

THE STEADY STATE SOLUTION

The solution is given for an unbounded queue (i.e. $K \leq L < \infty$) as well as a bounded queue (i.e. finite $L \geq K$).

Following the spectral expansion solution, the steady-state probabilities of the system considered can be expressed as:

$$p_{i,j} = \lim_{t \to \infty} P(I(t) = i, J(t) = j);$$
$$0 \leq i \leq N, \quad 0 \leq j \leq L$$

where L can be finite or infinite. Let's define diagonal matrices of size (N+1)x(N+1) as:

$$D_j^A(i,i) = \sum_{k=0}^{N} A_j(i,k);$$
$$D_j^B(i,i) = \sum_{k=0}^{N} B_j(i,k);$$
$$D_j^C(i,i) = \sum_{k=0}^{N} C_j(i,k);$$
$$D^A(i,i) = \sum_{k=0}^{N} A(i,k);$$
$$D^B(i,i) = \sum_{k=0}^{N} B(i,k);$$
$$D^C(i,i) = \sum_{k=0}^{N} C(i,k);$$

and

$$Q_0 = B, \quad Q_1 = A - D^A - D^B - D^C, \quad Q_2 = C.$$

These matrices are used in the spectral expansion solution for both bounded and unbounded queuing systems.

Unbounded Queuing System

For an unbounded queuing system, all state probabilities in a row (the row vectors, $\boldsymbol{v}_j$) can be defined as:

$$\mathbf{v}_j = (p_{0,j}, p_{1,j}, \ldots, p_{N,j}); \quad j = 0,1,2,\ldots$$

The steady-state balance equations can now be written as:

$$\mathbf{v}_0\left[D_0^A + D_0^B\right] = \mathbf{v}_0 A_0 + \mathbf{v}_1 C_1 \qquad (2)$$

$$\mathbf{v}_j\left[D_j^A + D_j^B + D_j^C\right] = \mathbf{v}_{j-1} B_{j-1} + \mathbf{v}_j A_j + \mathbf{v}_{j+1} C_{j+1}; \quad 1 \le j \le M-1 \qquad (3)$$

$$\mathbf{v}_j\left[D^A + D^B + D^C\right] = \mathbf{v}_{j-1} B + \mathbf{v}_j A + \mathbf{v}_{j+1} C \quad j \ge M \qquad (4)$$

and the normalizing equation:

$$\sum_{j=0}^{\infty} \mathbf{v}_j e = \sum_{j=0}^{\infty}\sum_{i=0}^{N} p_{i,j} = 1.0$$

from Equation (4) one can write

$$\mathbf{v}_j Q_0 + \mathbf{v}_{j+1} Q_1 + \mathbf{v}_{j+2} Q_2 = 0; \quad j \ge M-1$$

Furthermore, the *characteristic matrix polynomial $Q(\lambda)$* can be defined as:

$$Q(\lambda) = Q_0 + Q_1\lambda + Q_2\lambda^2$$

where

$$\psi Q(\lambda) = 0; \quad \left|Q(\lambda)\right| = 0.$$

λ and ψ are eigenvalues and left-eigenvectors of $Q(\lambda)$ respectively. Note that, ψ is a vector and

$$\psi = \psi_0, \psi_1, \ldots, \psi_N$$
$$\lambda = \lambda_0, \lambda_1, \ldots, \lambda_N$$

Finally, for an unbounded system, and avoiding large numbers resulting from the positive powers of eigenvalues greater than 1.0, one can obtain the general solution as:

$$\mathbf{v}_j = \sum_{k=0}^{N} a_k \psi_k \lambda_k^{j-M+1}, \quad j \ge M-1$$

and in the state probability form,

$$p_{i,j} = \sum_{k=0}^{N} a_k \psi_k(i) \lambda_k^{j-M+1}, \quad j \ge M-1$$

where a_k (k=0,1, …, N) are arbitrary constants which can be scalar or complex. The remaining $\boldsymbol{v}_j$ and a_k values can be obtained using an iterative process (Chakka 1995).

Bounded Queuing System

Analyses presented for the unbounded queue apply to the bounded queue with $0 \le j \le L$ with the balance equations given as follows:

$$\mathbf{v}_0\left[D_0^A + D_0^B\right] = \mathbf{v}_0 A_0 + \mathbf{v}_1 C_1 \qquad (5)$$

$$\mathbf{v}_j\left[D_j^A + D_j^B + D_j^C\right] = \mathbf{v}_{j-1} B_{j-1} + \mathbf{v}_j A_j + \mathbf{v}_{j+1} C_{j+1}; \quad 1 \le j \le M-1 \qquad (6)$$

$$\mathbf{v}_j\left[D^A + D^B + D^C\right] = \mathbf{v}_{j-1} B + \mathbf{v}_j A + \mathbf{v}_{j+1} C \quad M \le j < L \qquad (7)$$

$$\mathbf{v}_L\left[D^A + D^C\right] = \mathbf{v}_{L-1} B + \mathbf{v}_L A \qquad (8)$$

The normalizing equation is given as:

$$\sum_{j=0}^{L} \mathbf{v}_j e = \sum_{j=0}^{L}\sum_{i=0}^{N} p_{i,j} = 1.0$$

From Equation (7)

$$\mathbf{v}_j Q_0 + \mathbf{v}_{j+1} Q_1 + \mathbf{v}_{j+2} Q_2 = 0; \quad (M-1) \le j \le (L-2)$$

and the *characteristic matrix polynomials* can be expressed as:

$$Q(\lambda) = Q_0 + Q_1\lambda + Q_2\lambda^2$$

$$\bar{Q}(\beta) = Q_2 + Q_1\beta + Q_0\beta^2$$

where

$$\psi Q(\lambda) = 0; \quad \left|Q(\lambda)\right| = 0$$

$$\phi \bar{Q}(\beta) = 0; \quad \left|\bar{Q}(\beta)\right| = 0$$

β and ϕ are eigenvalues and left-eigenvectors of $\bar{Q}(\beta)$ respectively. Note that, ϕ is a vector and

$$\phi = \phi_0, \phi_1, \ldots, \phi_N$$
$$\beta = \beta_0, \beta_1, \ldots, \beta_N$$

Furthermore,

$$\mathbf{v}_j = \sum_{k=0}^{N} (a_k \psi_k \lambda_k^{j-M+1} + b_k \phi_k \beta_k^{L-j}), \quad M-1 \le j \le L$$

and in the state probability form,

$$p_{i,j} = \sum_{k=0}^{N} (a_k \boldsymbol{\psi}_k(i) \lambda_k^{j-M+1} + b_k \phi_k(i) \beta_k^{L-j}),$$
$$M-1 \le j \le L$$

where b_k (k=0,1, …, N) are arbitrary constants which can be scalar or complex.

We have solved the balance equations for both cases (Equations (2) – (8)), using the spectral expansion method, computed the state probabilities and obtained the mean queue length as:

$$MQL = \sum_{j=0}^{L} j \sum_{i=0}^{N} p_{i,j}$$

where L can be finite or infinite depending on whether the case concerned is bounded or unbounded.

NUMERICAL RESULTS

To show the effectiveness of the method presented and evaluate the performance of the proposed system, we first considered 1, 2, 3, and 4-processor systems with break-downs and an infinite queue. Other parameters are given as σ jobs/sec, ξ=0.01, η=0.5, μ=4000 jobs/hr, φ = 2 /hr, and δ = 60 /hr unless stated otherwise.

Figure 3 shows the relationship between the mean queue length and the mean arrival rate σ, for different number of servers and c=0 (a single-server system is independent of c).

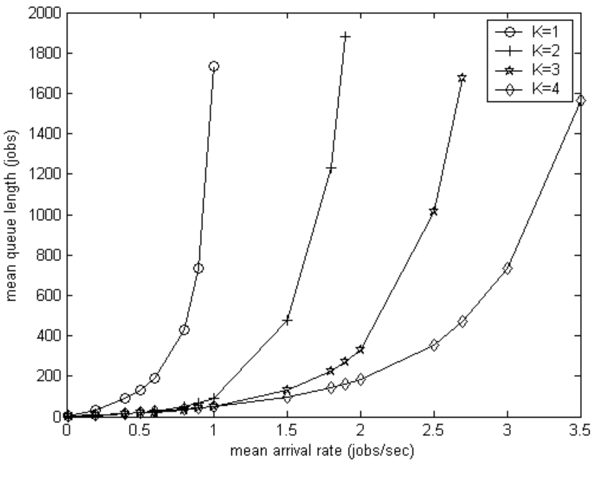

Figure 3: Mean Queue Length versus Mean Arrival Rate

Figure 4 shows the mean queue length as a function of c. Other parameters are σ = 20 jobs/sec, σ/(Kμ)=0.7, ξ=0.01, η=0.5, φ = 10 /hr, and δ = 50 /hr. It is clearly evident that an increase in c results in a decrease in the mean queue length because reconfiguration delays are shorter than rebooting delays. Here, it is important to note that when σ/(Kμ) is kept constant, the 3-processor system considered performs better than the 4-processor system specified, especially as c increases. This is because failure rate is proportional to the number of operational processors and in case of a single processor failure, the whole system goes down for a period of 1/δ or 1/φ as appropriate.

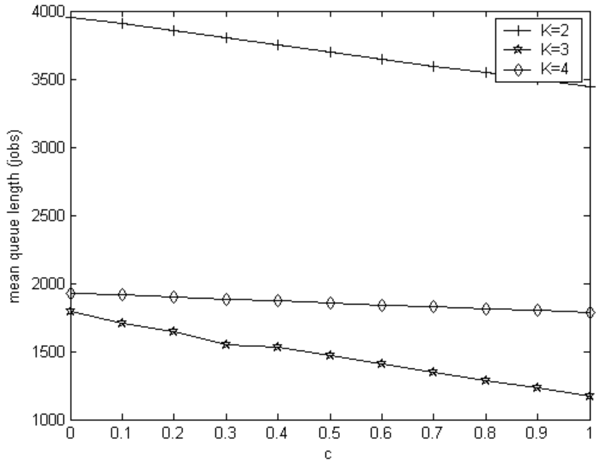

Figure 4: Mean Queue Length as a Function of c for 2, 3, and 4-Processor Systems

The parameters 1/δ or 1/φ are certainly important in determining the system degradation due to reconfiguration/rebooting delays and identifying the optimum number of processors. We computed the mean queue length as a function of reconfiguration rate δ for K=3. σ = 1 job/sec, σ/(Kμ)=0.7, ξ=0.01, η=0.5, and φ = 2 /hr. This is illustrated in Figure 5. The results indicate that as c increases the change in the mean queue length decreases considerably. This shows that φ plays an important role in degrading the system's performance.

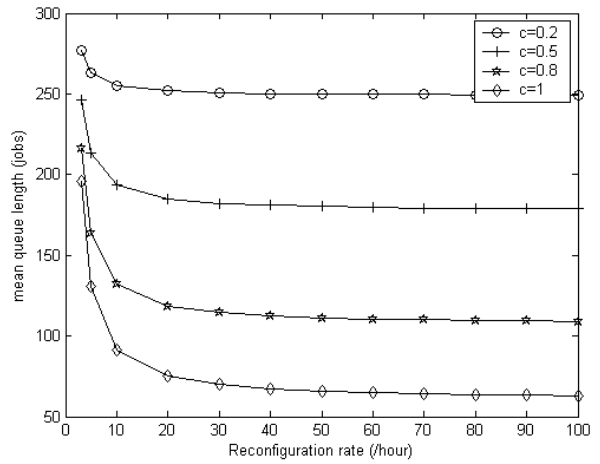

Figure 5: Mean Queue Length as a Function of δ and c

Figure 6 shows mean queue length as a function of c, with K = 1, 2, and 3, σ = 20 jobs/sec, and σ/(Kμ) = 0.7. This shows that depending on the values of reconfiguration and rebooting delays, the mean queue length performance of a 2-processor system may become better than that of a 4-processor system for some c values and approaches to the performance of a 3-processor system. The performance of a single processor system does not depend on c and the performance of such a system is shown for comparison purpose only.

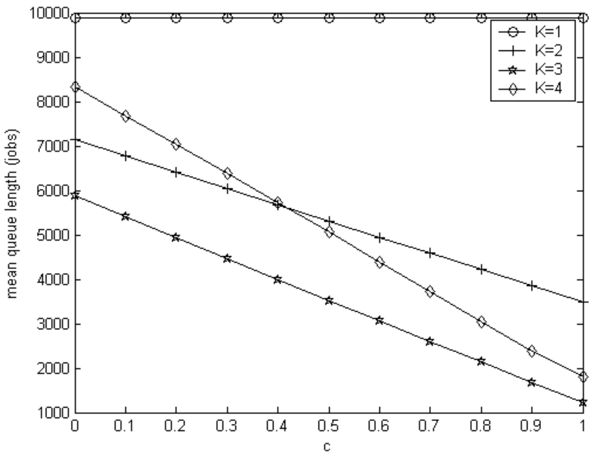

Figure 6: Mean Queue Length as a Function of c

Figure 7 shows how mean queue length decreases as c increases for K=3, σ = 1 job/sec, and $\sigma/(K\mu)$ = 0.7. The corresponding mean queue length values are presented for various values of δ. Again, for larger reconfiguration delays (i.e. δ small) performance degradation is evident even at higher c values. As the reconfiguration delay decreases (e.g. δ > 20) the degradation is mainly due to c values.

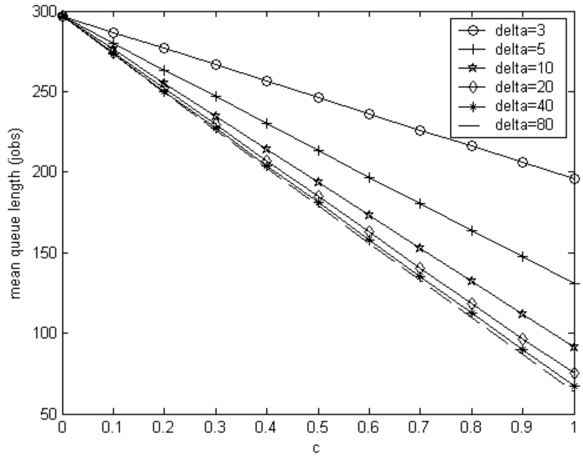

Figure 7: Mean Queue Length as a Function of c and δ

Figure 8 shows the probability that the system is idle, for various K and c values with σ = 20 jobs/sec, and $\sigma/(K\mu)$ = 0.7. Here, different c values did not seem to matter.

To demonstrate the effects of finite queuing capacity, mean queue length of a bounded system is calculated. First, Figure 6 has been reproduced for L = 100, and σ = 1 job/sec. Figure 9 shows MQL as a function of c for finite L. All other parameters are same as the ones used for Figure 6. Clearly, when $\sigma/(K\mu)$=0.7, L is the limiting factor, and K has a negligibly small effect. Similarly, Figure 7 was reproduced for L = 300. Again, the limiting factor here is L. This is illustrated in Figure 10.

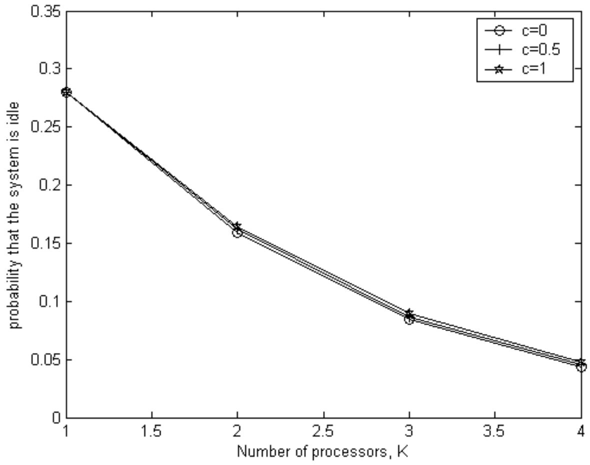

Figure 8: p_0 as a Function of the Number of Servers with $\sigma/(K\mu)$=0.7 and Various c Values

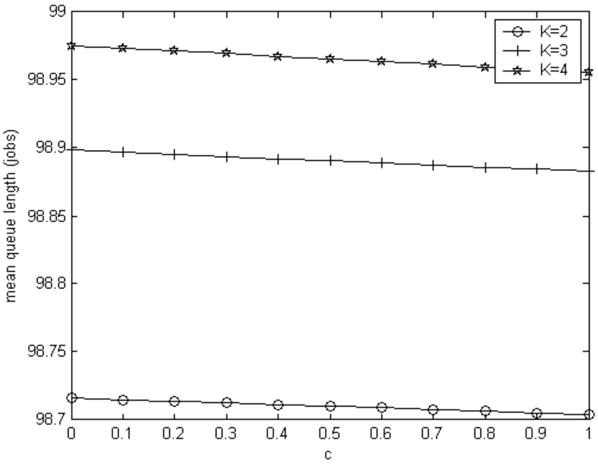

Figure 9: Mean Queue Length as a Function of c for 1, 2, 3, and 4-Processor Systems and L = 100

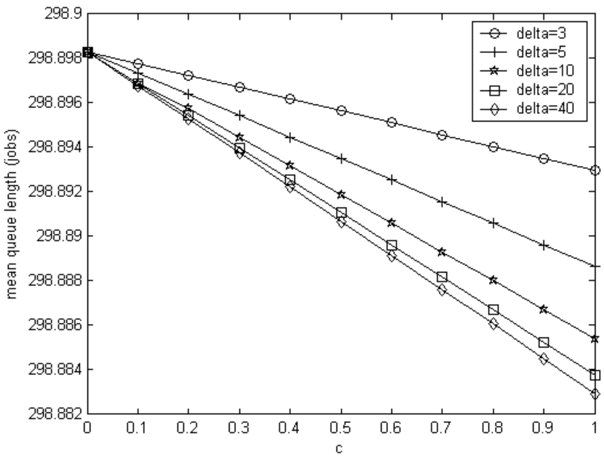

Figure 10: Mean Queue Length as a Function of c and δ for L = 300

Finally, the mean queue length and job loss rate are calculated for various σ values with K=3 and 4, c=0, 0.5, and 1, μ=2 jobs/sec, ξ=0.01, η=0.5, $1/\varphi$ = 500 sec,

$1/\delta$ = 50 sec, and L = 200. Figure 11 and Figure 12 illustrate the results obtained.

As it can be seen on Figure 11, the effect of c on mean queue length is significant for smaller σ values for K=3 as well as K=4. This is due to the relatively low demand on resources. However, as σ increases, L becomes the main limiting factor in the performance of the system and the mean queue length approaches L. K=4 will only be preferable over K=3 if good cover is provided (i.e. c=1) and σ is relatively large.

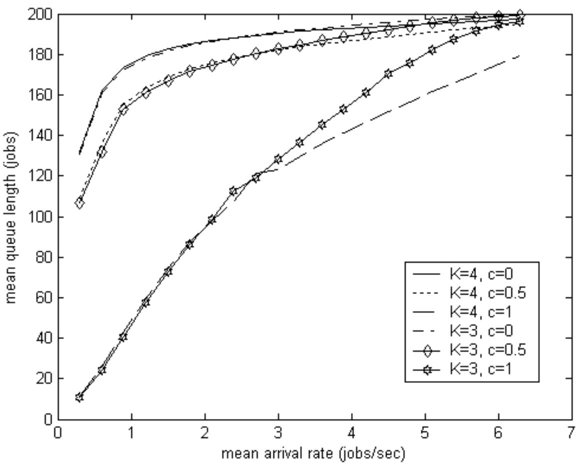

Figure 11: Mean Queue Length as a Function of K, c, and σ for L = 200

Figure 12 shows that, when the percentage of jobs lost is considered, c becomes an important parameter for all σ values. For larger σ values, L becomes an important limiting factor, especially for low c values.

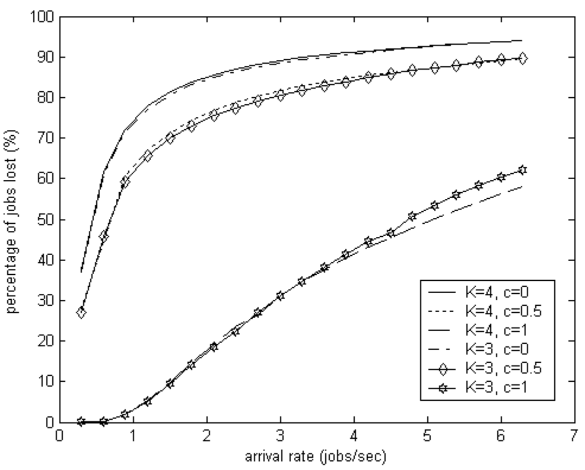

Figure 12: Percentage of Jobs Lost as a Function of K, c, and σ for L = 200

CONCLUSIONS AND RECOMMENDATIONS

In this paper multiprocessor systems with break-downs, and, reconfiguration and rebooting delays have been modelled for exact solution. The state probabilities in the case of a homogeneous multiprocessor system with breakdowns, repairs, reconfiguration and rebooting delays are derived using the spectral expansion method. Numerical results have been obtained and presented for various performability parameters, for both bounded and unbounded systems. Results show that, when queue limit is not an important factor on mean queue length performance, the choice of the optimum number of processors depends on the values of $1/\delta$, and $1/\varphi$ as well as c and demonstrate the effect of these parameters on system performance. However, for bounded queuing systems, L is the main factor affecting the mean queue length performance of the system at relatively large σ values.

The method can be extended to the case of heterogeneous multiprocessor systems with non-identical servers and also to many of the high performance/highly available/highly reliable computer architectures. The performability of flexible manufacturing cells can also be modelled using the method presented and hence the model becomes highly relevant to manufacturing or production research.

REFERENCES

Buzacott, J. A. and J.G. Shanthikumar, 1993. *Stochastic Models of Manufacturing Systems*, Prentice-Hall.

Chakka, R. and I. Mitrani, 1992. "A numerical solution method for multiprocessor systems with general breakdowns and repairs". In *Proceedings of the 6th International Conference Modelling Techniques and Tools*, (Edinburgh, UK), 289-304.

Chakka, R. and I. Mitrani, 1994. "Heterogeneous multiprocessor systems with breakdowns: performance and optimal repair strategies", *Theoretical Computer Science*, 125, 91-109.

Chakka, R., 1995. *Performance and Reliability Modelling of Computing Systems Using Spectral Expansion*, PhD Thesis, University of Newcastle Upon Tyne, UK.

Chakka, R. 1998. "Spectral expansion solution for some finite capacity queues", *Annals of Operations Research*, 79, 27-44.

Chakka, R., O. Gemikonakli and P. Basappa, 2002. "Modelling Multiprocessor Systems With Time or Operation Dependent Breakdowns, Alternate Repair Strategies, Reconfiguration and Rebooting Delays". In *Proceedings of SPECTS 2002*, (San Diego, CA, July).

Fiems, D., B. Steyaert, and H. Bruneel, 2004. "Discrete-time queues with generally distributed service times and renewal type server interruptions", *Performance Evaluation*, 55, 277-298.

Harrison, P. G. and N.M. Patel, 1993. *Performance Modelling of Communication Networks and Computer Architecture*, Addison-Wesley.

Stecke, K. E. and I. Kim, 1989. "Performance evaluation for systems of pooled machines of unequal sizes: unbalancing vs. blocking". *European Journal of Operations Research* 42, 22-38.

Stecke, K.E. 1992. "Machine interference: Assignment of machines to operators". *Handbook of Industrial Engineering, Second Edition*, Chapter 57, Gavriel Salvendy (Ed.).

Righter, R. 1996. "Optimal Policies for Scheduling Repairs and Allocating Heterogeneous Processors". *Journal of Applied Probability*, 33, 536-547.

Trivedi, K. S., 2002. *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*, Wiley, NY, USA.

Trivedi, K. S. A.S. Sathaye, O.C. Ibe and R.C. Howe, 1990. "Should I add a processor?", In *Proceedings of 23rd Annual Hawaii International Conference on System Sciences*, IEEE Computer Society Press, (Los Alamos, CA), 214-221.

AUTHOR BIOGRAPHIES

ORHAN GEMIKONAKLI was born in Yesilyurt, Cyprus, and went to Eastern Mediterranean University in Cyprus, where he studied electrical engineering and obtained his degree in 1984. He then continued his studies at King's College, University of London where he obtained his MSc in Digital Electronics, Computers and Communications and PhD in Telecommunications in 1985 and 1990 respectively. He worked there as a post-doctoral Research Associate for a couple of years before moving in 1990 to Middlesex University, London where he is now a Principal Lecturer leading postgraduate studies in Computer Communications. He is a member of the IEE, a member of the IEEE and a Chartered Engineer. His research interest is in network security and performance modelling. His e-mail address is: o.gemikonakli@mdx.ac.uk and his Web-page can be found at http://www.cs.mdx.ac.uk/staffpages/orhan/.

TIEN VAN DO obtained a PhD degree from the Budapest University of Technology and Economics in 1996. He is an associate professor at Department of Telecommunications, Budapest University of Technology and Economics and a leader of Networking Technology Laboratory. His research interests include network technology, performance evaluation and network design.

RAM CHAKKA received his B.Engg. (Electrical and Electronics, 1980), M.S.(Engg) (Computer Science, 1986) from the Indian Institute of Science, Bangalore, India and Ph.D. (Computer Science, 1995) from the University of Newcastle upon Tyne, UK. Presently he is a Professor and Head of the RDQ (Research, Development, Quality of education) unit at the RGMCET, Nandyal, India. Earlier, he worked at Indian Institute of Science, University of Newcastle upon Tyne, Imperial College (London), Middlesex University (UK), Norfolk State University (USA) and Sri Sathya Sai Institute of Higher Learning (India). He published a number of papers in Performability Modelling.

ENVER EVER was born in Finsbury, London. He obtained his BSc. degree from the Department of Computer Engineering, Eastern Mediterranean University, Cyprus in 2002 and his MSc in Computer Networks from Middlesex University, London in 2003. Currently he is a PhD student at Middlesex University. His research is on mathematical modelling for performance evaluation of various multiserver systems. His research interest is in networks and performance modelling. His email address is: e.ever@mdx.ac.uk.