APPROXIMATE ANALYSIS OF MARKOVIAN QUEUEING NETWORKS WITH PRIORITIES

W. Schmitt

Dept. of Communications, University of Siegen
Siegen, Federal Republic of Germany

ABSTRACT

Modelling of data communication networks and computer systems, for purposes of performance evaluation, often leads to multi-class queueing networks with priorities. Whereas for single stage queueing systems with priorities numerous publications have appeared in the past, only little is known on the influence of priorities in queueing networks.

In this paper we give various new approximation techniques for analysing open and closed Markovian queueing networks with priority classes. The methods are based on establishing 'Virtual Server' models with state dependent reduced service rates. These models provide an appropriate description of the behaviour of the considered class and lead to single class queueing networks which can be analysed by well known techniques. Numerical results are given and validated either by exact calculations or by simulations.

1. INTRODUCTION

Priority disciplines are used in a variety of communication and computer systems. For example, one considers a packet switching network with three types of messages: network control messages, time critical messages and deferrable messages. Then network control will have priority over time critical messages and these over deferrable work.

Another example is a computer system with multi-programming and priority scheduling disciplines at the central processing unit (CPU). The modelling of such systems, with the object of performance analysis, leads in the first case to an open and in the second case to a closed queueing network with priority classes. Whereas for single-stage queueing systems with priorities numerous publications have appeared in the past, only little is known on the behaviour of priority queueing networks. There are very few exact results to be found in the literature (see Refs. (1), (6), (7), (10), and (14)) and approximate results as well (see Refs. (4), (7), (8), (12), (13), and (15)). Most of these approaches are restricted to very special models. With the exception of (6) always Markovian properties are supposed.

In this paper we discuss various new approximation techniques to analyse open and closed Markovian queueing networks with priorities. The methods are based on the concept of 'Virtual Server' models which has already been applied in (7), (12), and (13). In these publications low priority customers are assumed to be served effectively by a Virtual Server whose service rate is that of the real processor reduced by an appropriate chosen factor. In contrast to these Virtual Servers with state independent reduced service rates we derive server models with state dependent reduced service rates for preemptive and nonpreemptive priority disciplines. These models provide a correct description of the corresponding random processes of system states in the single stage queueing systems M/M/1-PRE and M/M/1-NONPRE as well as in the cyclic queueing system */M/1-PRE + */M/1-PRE. Unfortunately, the effective service rates for low priority customers are, just as in (7), a-priori unknown and our approach here is to approximate them in an appropriate manner. This leads to a quite simple and intuitive algorithm for analysing open queueing networks with preemptive and/or nonpreemptive priorities. But for closed queueing networks, where only preemptive priorities are considered, a more complex iterative technique is necessary. Numerical results are given for both cases and validated either by exact calculations or by simulations. Additionally we compare our approach with the methods proposed in (7) and (12). Relations between these various techniques are pointed out. Although only two priority classes are considered throughout this paper, an extension to multiclass structures is possible using the concept of class composition (4).

2. THE 'VIRTUAL SERVER' CONCEPT

The 'Virtual Server' concept is sketched in Fig. 1, where:

\[ A_r(t) = 1 - \exp(-\lambda_r t) \]

interarrival time distribution function

\[ H_r(t) = 1 - \exp(-\mu_r t) \]

service time distribution function

\[ \lambda_r \]

arrival rate

\[ \mu_r \]

service rate

\[ h_r = 1/\mu_r \]

mean service time

\[ w_r \]

mean waiting time

\[ r = 1 \]

high priority class

\[ r = 2 \]

low priority class
marks the corresponding symbols of the Virtual Server models.

**PRIORITY SYSTEM**

\[ A_1(t) \quad A_2(t) \]

\[ \lambda_1 \quad \lambda_2 \]

\[ w_1 \quad w_2 \]

**VIRTUAL SERVER** MODELS

\[ h_1(t) \quad h_2(t) \]

\[ h_1 \quad h_2 \]

**Fig. 1:** Decomposition of a priority system into Virtual Server models.

The basic idea is to assume that each class receives service from a dedicated Virtual Server, whose service time distribution function is chosen such that the priority mechanism is appropriately taken into account. Replacing the original server by these Virtual Servers, a two-class queueing network will be decomposed into two single-class networks which are to be analysed by known methods.

Concerning preemptive resume priorities, where it is sufficient to consider the low priority class only, we will give two known examples for such Virtual Server models. The first one is drawn from (12) and represented in Fig. 2a. Therein, the low priority customers have a dedicated exponential server with a constant reduced service rate \( \lambda_2 \). (1-\( \rho_1 \)), where \( \rho_1 \) is the utilization of the first class. That is, the low priority customers receive the remaining work capacity of the real server. Another motivation, leading to this choice of \( \lambda_2 \), is that \( 1/\mu_2 \) is equal to the mean completion time \((6)\). The completion time \( T_C \) is the period which elapses between the instant a customer enters the server for the first time and the instant he has finished his service completely. But, if a low priority customer is ignorant of the fact that his service is being interrupted by higher priority customers \( T_C \) will not be the time he perceives his service time to be. Kaufman (7) has proved that the average \( h_2^* \) of this fictitious service time is given by

\[ h_2^* = \frac{\text{prob}(X_2>0)}{\text{prob}(X_1=0, X_2>0)} \cdot h_2, \quad (1) \]

where \( X_2 \) denotes the number of customers in system of class \( r \) \((r=1,2)\). This observation has led to the Virtual Server model shown in Fig. 2b. The obvious problem there is, that the probabilities in eq. (1) are a-priori unknown and must be approximated \((8)\).

Since in both examples the service rate of the Virtual Server has been reduced by a constant factor, we follow the suggestion in \((7)\) and denote the first approach the reduced occupancy approximation (roa) and the second one the modified reduced occupancy approximation (m-roa).

### 3. STATE DEPENDENT SERVER MODELS

#### 3.1 Preemptive Priorities

In this section we derive a new Virtual Server model for the second class in a Markovian queueing system with preemptive priorities (resume or repeat with resampling).

Let \( p(m,n) = \text{prob}(X_1=m, X_2=n) \) be the stationary distribution of the state \((m,n)\), where \( m,n=0,1,2,\ldots \). Then, \( p(m,n) \) satisfies the stationary balance equations (see Ref. \((16)\)):

\[ \text{2a:} \quad \text{2b:} \quad \text{2c:} \quad \text{2d:} \]

with \( \lambda = \lambda_1 + \lambda_2 \).

Summation over \( m \) yields after some algebraic manipulations

\[ \lambda_2 p_2(n) = \lambda_1 p_2(n) - \lambda_2 p_2(n+1) \quad n>0 \quad (4a) \]

\[ \lambda_2 p_2(0) = \mu_2 p_2(1) \quad (4b) \]

where \( p_2(n) = \frac{1}{\lambda} p(m,n) \quad m=0 \quad (4c) \)

defines the marginal distribution for the number of second class customers in system. Setting

\[ \mu_2^*(n) = \frac{\text{prob}(X_1=0, X_2=n)}{p_2(n)} \quad (5a) \]

\[ c = \frac{\text{prob}(X_1=0, X_2=n)}{p_2(n)} \quad (5b) \]

the equations \((4a,d)\) become

\[ \lambda_2 p_2(n) - (\lambda_2 + \mu_2^*(n)) p_2(n) + \mu_2^*(n+1) p_2(n+1) = 0 \quad (6a) \]

\[ -\lambda_2 p_2(0) + \mu_2^*(1) p_2(1) = 0 \quad (6b) \]

These are the balance equations for a \( M/M/1 \) queueing system with state dependent service rates \( p_2(n) \). Consequently, we have a Virtual Server model which provides an exact description of the marginal distribution \( p_2(n) \) of the queueing system \( M/M/1-PRE \). It should be mentioned however that, concerning the global behaviour of the traffic flow of low priority customers, our approach is only an approximation. Especially the output stream of the second class is not a Poisson process in general. But all mean values obtained by means of the \( p_2^*(n) \) are exact. We denote our method the state dependent reduced occupancy approximation (sd-roa) and state that the m-roa turns into the sd-roa if in equation \((1)\) the expression \( X_2>0 \) is replaced by \( X_2=n \).
An interesting quantity is the weighted sum

\[ c_2 = \sum_{n=1}^{\infty} c_n p_2(n) = \rho_2 \tag{7} \]

\( h_2/c_2 \) plays the role of a virtual mean service time. For \( P_1 + P_2 = 1 \) eq. (7) becomes \( c_2 = 1 - \rho_1 \). That is, only in this case a correct description of the mean service time a low priority customer perceives is provided by the roa. This observation agrees with a result of the investigations made in (7).

Fig. 2 gives an overview of the three different Virtual Server models we have discussed here. Our final remark in this section is that the coefficient \( c_n \) are unknown a-priori. As we shall see in chapter 4 this will cause no problems in case of open queueing networks. But for closed queueing networks an approximation method to determine the \( c_n \)'s will be necessary.

- roa:
  \[ \mu_2^* = (1-\rho_1) \mu_2 \]
- m-roa:
  \[ \mu_2^* = \frac{\text{prob}(X_1=0,X_2>0)}{\text{prob}(X_2>0)} \mu_2 \]
- sd-roa:
  \[ \mu_2^* = \frac{\text{prob}(X_1=0,X_2=n)}{\text{prob}(X_2=n)} \mu_2 \]

Fig. 2: Comparison of the different reduced service rates for the roa, m-roa, and the sd-roa.

### 3.2 Nonpreemptive Priorities

For the M/M/1-NONPRE queueing system an exact description of the marginal distributions \( P_1(n) \) and \( P_2(n) \) is provided by M/M/1 queueing systems with state dependent service rates again. The expressions for \( \mu_1^*(m) \) and \( \mu_2^*(n) \) are:

\[ \mu_1^*(m) = c_1(m) \cdot \mu_1 = \sum_{n=0}^{\infty} p(m,n,1) \cdot \mu_1, \text{ m>0} \tag{8a} \]
\[ \mu_2^*(n) = c_2(n) \cdot \mu_2 = \sum_{m=0}^{\infty} p(m,n,2) \cdot \mu_2, \text{ n>0} \tag{8b} \]

which can be rewritten into

\[ \mu_1^*(m) = c_1(m) \cdot \mu_1 = 1 - \frac{\text{prob}(X_1=m,X_2>0,r=2)}{\text{prob}(X_1=m)} \mu_1 \tag{8c} \]
\[ \mu_2^*(n) = c_2(n) \cdot \mu_2 = 1 - \frac{\text{prob}(X_2>0,X_2=n,r=1)}{\text{prob}(X_2=n)} \mu_2 \cdot \tag{8d} \]

The variable \( r \) defines whether there is a high (r=1) or a low (r=2) priority customer in service.

To prove these results one considers the balance equations for the M/M/1-NONPRE queueing system (11) from which the balance equations for the marginal distributions \( P_1(m) \) and \( P_2(n) \) are obtained by summations over \( n \) and \( m \) respectively. The next step comprises some simple manipulations, namely the multiplication of the appearing sums \( \Sigma p(m,n,1) \) and \( \Sigma p(m,n,2) \) with \( p_1(m)/p_1(m) \) and \( p_2(n)/p_2(n) \) respectively. After this we can recognize the service rates of the equivalent state dependent M/M/1 queueing systems directly.

If we compare the formulas (8c,d) with the corresponding expressions in (7) we will see once more that the m-roa turns into the sd-roa if the expressions \( X_1>0 \) and \( X_2>0 \) will be replaced by \( X_1=m \) and \( X_2=n \). Finally we note:

\[ c_1(m) = \sum_{m=1}^{\infty} p(m) \cdot c_1(m) = \rho_1 \tag{9a} \]
\[ c_2(n) = \sum_{n=1}^{\infty} p(n) \cdot c_2(n) = \rho_2 \cdot \tag{9b} \]

### 4. Analysis of Open Queueing Networks

#### 4.1 Outline of the Analysis Principle

In this chapter we are concerned with priority disciplines in open queueing networks. Specifically, there are a number of *M/1-PRE* and/or *M/1-NONPRE* queueing stations (nodes) which are interconnected arbitrarily for each class. Supposing external input streams to be Poisson, the network is defined through the following parameters:

- \( N \): Total number of queueing stations
- \( \lambda_{ir} \): Vector of external arrival rates for class \( r \)
- \( \mu_{ir} \): Vector of mean service times for class \( r \)
- \( Q_{ir} \): Routing matrix for class \( r \)

The total arrival rates \( \lambda_{ir} \) of queueing station \( i \) are obtained from the following equations representing the conservation of flow:

\[ \lambda_{ir} = \lambda_{oir} + \sum_{j=1}^{N} \lambda_{jr} Q_{ijr}, \text{ i=1,2,...,N; r=1,2} \tag{10} \]

Applying our state dependent server models the network will be decomposed into two single class networks with product form solutions (2). Hence, it follows a simple procedure for analysing open queueing networks with priorities:

**Step 1:** Compute the throughputs \( \lambda_{ir} \) of each node from equation (10)

**Step 2:** Analyse each queueing station (see Refs. (5), (16)) separated from the network and subjected to Poisson arrivals with the rates \( \lambda_{ir} \), obtained from Step 1.

As pointed out in section 3.1, the second step is only an approximation. A generalization of the algorithm for open networks with more than two priority classes is straightforward.
4.2 Numerical Results and Validations

To prove the qualification of the approximation method we have applied it to two different test-bed network models. The first one is a serial queueing system $M/M/l$-NONPRE $\rightarrow \cdot /M/l$-NONPRE which has also been analysed through solving the balance equations by using Gauss-Seidel iteration. A few results are contrasted in Table 1. We see that the amount of the maximum relative error is 4.4%.

Our second test-bed network model consists of four nodes and is defined through the following parameters:

Test-bed network model II

$N = 4$

$\lambda_{01} = (0.01, 0.01, 0.01, 0.04)^T$

$\lambda_{02} = (0.07, 0.07, 0.07, 0.1)^T$

$Q_0 = Q_2 = \begin{bmatrix}
0.2 & 0.3 & 0.2 \\
0.2 & 0.3 & 0.2 \\
0.3 & 0.3 & 0.3 \\
0.2 & 0.3 & 0.3 \\
\end{bmatrix}$

Case 1): $h_1 = (1, 1, 1, 1.2)^T$; $h_2 = h_1$

Case 2): $h_1 = (1, 1, 1, 1.2)^T$; $h_2 = 0.5 \cdot h_1$

Case 3): $h_1 = (0.5, 0.5, 0.5, 0.6)^T$; $h_2 = 2 \cdot h_1$

Remarks: $T$ denotes the transposition operator $q_{j0r} = \text{prob}(\text{class } r \text{ customer leaves at node } j)$

The parameters have been chosen such that there is a bottleneck at node 4. Results are shown in Fig. 3 where in case a) preemptive priorities and in case b) nonpreemptive priorities are assumed for all nodes. We see that all curves are in a good accordance with the simulations.

5. ANALYSIS OF CLOSED QUEUEING NETWORKS

5.1 Outline of the Analysis Principle

In this chapter we will deal with preemptive (resume or repeat with resampling) priority disciplines in closed queueing networks. In addition to our notations for open queueing networks we define:

$M_r$ Total number of class $r$ customers inside the network.

Since there are no exogenous arrivals in closed queueing networks, the flow balance equations (10) change into:

$\lambda_{ir} = \sum_{j=1}^{N} \lambda_{jr} q_{jir}, \quad i=1,2,\ldots,N; \quad r=1,2.$ \hspace{1cm}(11)

Thus, the throughputs $\lambda_{ir}$ are only determined up to arbitrary constant factors through these systems of homogenous linear equations. We mention that the

![Fig. 3: Mean flow time versus total exogenous arrival rate at station 4](image)

**Fig. 3**: Mean flow time versus total exogenous arrival rate $\lambda_{04} - \lambda_{041} - \lambda_{042}$ for the test-bed network model II. Simulation results (95% confidence intervals).
The notion of throughput is commonly used in context with closed queueing networks where throughput of station i and arrival rate at station i are equivalent. The fact that only the relative throughputs can be obtained from the eqs. (11) causes difficulties in approximating the coefficients $c_{ni}$ defined by eq. (5b). Before giving a method to approximate these quantities we will have a short look at the cyclic queueing system $*/M/1$-PRE.

The cyclic queueing system $*/M/1$-PRE has been analyzed comprehensively in (10). Concerning this model we have made the following basic observation: If we calculate the coefficients $c_{ni}$ using the exact probabilities and subsequently the BCMP theorem (2) to the corresponding network with state dependent Virtual Servers the numerical results from (10) will be exactly reproduced. Especially, this observation motivates to apply our state dependent server models to analyze closed queueing networks with priorities. Comparing results achieved by means of the m-roa and the sd-roa it is shown in Table 2 that the latter technique provides a significant improvement in those cases where m-roa fails.

Subsequently, we will outline an approximation method to calculate the unknown coefficients $c_{ni}$ for closed priority networks with arbitrary interconnections for each class. The technique is best explained by reference to Fig. 4. In order to approximate the stationary state probabilities $p(m_i,n_j)$ for each node i, the state-transition-rate diagram shown in Fig. 4b has been developed.

For ease of representation we have chosen $M_1=M_2=2$. The transition rates $\mu_{ic}(k)$, where $k=1,2,\ldots,M_i$, are to ascertain applying Norton’s theorem for queueing networks (see Refs. (3), (9)) with respect to the first class only (Fig. 4a). The unknown coefficients $\lambda_2$ are to determine by a search-algorithm such that the following eq. (12a) is fulfilled at the station j with highest utilization of first class customers:

$$M_2 \sum_{n=0}^{\infty} c_{nj} p_{2}(n_j) = 1-p_{2}(1) \cdot \frac{p_{2}(0)}{p_{2}(0)} \cdot (12a,b)$$

Eq. (12a) is an extension of eq. (7). The reason to meet this condition at that node with highest utilization of first class customers is given for this station will cause the most influence on the throughputs of second class customers. Before we

---

### Table 1: Tandem queueing system $M/M/1$-NONPRE $+/M/1$-NONPRE. Comparison of exact with approximate results.

<table>
<thead>
<tr>
<th>Case</th>
<th>$\lambda_{011}$</th>
<th>$\lambda_{012}$</th>
<th>$h_{11}$</th>
<th>$h_{12}$</th>
<th>$h_{21}$</th>
<th>$h_{22}$</th>
<th>$f_1$ exact</th>
<th>$f_1$ approx.</th>
<th>rel. error</th>
<th>$f_2$ exact</th>
<th>$f_2$ approx.</th>
<th>rel. error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1</td>
<td>0.4</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>3.084</td>
<td>3.110</td>
<td>+0.84%</td>
<td>4.212</td>
<td>4.206</td>
<td>-0.14%</td>
</tr>
<tr>
<td>2</td>
<td>0.1</td>
<td>0.4</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
<td>0.5</td>
<td>2.408</td>
<td>2.440</td>
<td>+1.33%</td>
<td>1.680</td>
<td>1.635</td>
<td>-2.68%</td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>0.4</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
<td>0.5</td>
<td>2.709</td>
<td>2.750</td>
<td>+1.51%</td>
<td>2.349</td>
<td>2.246</td>
<td>-4.40%</td>
</tr>
<tr>
<td>4</td>
<td>0.1</td>
<td>0.4</td>
<td>0.5</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
<td>1.970</td>
<td>1.895</td>
<td>-3.81%</td>
<td>3.618</td>
<td>3.624</td>
<td>+0.17%</td>
</tr>
<tr>
<td>5</td>
<td>0.2</td>
<td>0.4</td>
<td>0.5</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
<td>2.073</td>
<td>1.998</td>
<td>-3.61%</td>
<td>3.980</td>
<td>3.992</td>
<td>+0.30%</td>
</tr>
<tr>
<td>6</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
<td>2.187</td>
<td>2.118</td>
<td>-3.16%</td>
<td>4.442</td>
<td>4.464</td>
<td>+0.50%</td>
</tr>
<tr>
<td>7</td>
<td>0.4</td>
<td>0.4</td>
<td>0.5</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
<td>2.315</td>
<td>2.248</td>
<td>-2.89%</td>
<td>5.041</td>
<td>5.076</td>
<td>+0.69%</td>
</tr>
</tbody>
</table>

---

**Fig. 4:** Isolated queueing station i.

a) Node i with composite queue for class 1 and residual network for class 2.

b) State-transition-rate diagram for the approximate random process of system states at node i. $M_1 = M_2 = 2$, $X_i =$ number of class i customers at node i.
describe the complete procedure to analyse closed queueing networks with preemptive priorities, we introduce the notation:

\[ \mathbf{v}_2^{(1)} = (v_{12}^{(1)}, \ldots, v_{N2}^{(1)}) \]

1st approximate value of \[ \mathbf{v}_2 \]

**Procedure:**

**Step 1:** Solve the network for class 1 only.

**Step 2:** Compute the service rates \[ \mu_{1c}(k) \] applying Norton's theorem for queueing networks to each node with respect to class 1 only.

**Step 3:** Approximate the coefficients \[ c_{ni} \].

(i) Solve the network for class 2 by means of the roa and set \[ v_{12}^{(1)} = \lambda_2 - \text{roa} \], \[ \lambda_2 = 0 \].

(ii) Solve the stationary balance equations corresponding to Fig. 4b and determine the coefficients \[ c_{ni}^{(1)} \] using eq. (5b).

(iii) If eq. (12a) will be fulfilled go to step 4. Otherwise set \[ v_{12}^{(1)} = \alpha v_{12}^{(1)} \] and go to (i). The factor \[ \alpha > 0 \] is to choose with respect to the following rule: If the sum in eq. (12a) will be less than \[ (\lambda_2 + \alpha \rho_j) \] or greater than \[ (\lambda_2 - \alpha \rho_j) \] then go to (i). It should be mentioned here that for reasons of brevity the search algorithm, outlined in Step 3, has been described somewhat simplified. In order to distinguish this approach from the sd-roa we will denote it the approximate sd-roa (asd-roa).

**5.2 Numerical Results and Validations**

The asd-roa has been applied to the cyclic queueing system \[ *\text{/M/1-PRE} + *\text{/M/1-PRE} \] and to the queueing network sketched in Fig. 5 (test-bed network models III and IV). Several results are shown in Table 2 and Fig. 6 together with those obtained by application of the roa. For validation, exact and simulation results are given too. We see that the asd-roa provides better results then the roa in almost all cases. Especially, for the parameters chosen in Fig. 6b the asd-roa yields a significant improvement.

**CONCLUSIONS**

We have suggested new approximation techniques to analyse Markovian queueing networks with priorities. The methods are based on establishing Virtual Server models with state dependent service rates. Concerning open queueing networks, where the priority discipline may be preemptive and/or nonpreemptive, our approach yields numerical results which are in good accordance with simulations. For closed queueing networks, the proposed procedure is restricted to preemptive priorities only. Although the deviation of the numerical results from simulations and exact calculations may be considerable.
in some cases, especially then a significant improvement of the results achieved by means of the roa technique is provided.

Our further attention will be directed towards improving the asd-roa technique and extending it to nonpreemptive priority disciplines. Furthermore, we want to consider queueing network models in which the priority dispatching rule will not be present at each station.

ACKNOWLEDGEMENTS

The author would like to thank Prof. P. J. Kuehn for helpful discussions during the course of this work. He also thanks Mr. M. Schneider who has done the computations for the M/M/1-NONPRE → *M/1-NONPRE queueing system. The efforts of Mr. H. van As for supporting the simulations are greatly appreciated.

REFERENCES

Summary of Questions/Answers

Date: 09 June 1983
Session: 1.3
Paper: 3

Q.1 (Werner Bux)
The approximation results given in your paper show excellent agreement with simulation. Could you comment on whether this is true for all types of networks, or, if not, where possible limitations of your methods, regarding their accuracy, are.

A.1 (W. Schmitt)
In this paper two special types of priority queuing networks have been considered, namely open priority queuing networks with preceptive and/or non preceptive priorities at each node and closed priority queuing networks with a preemptive priority at each node*. For this type of network calculations are always in a good, sometimes very good, agreement with simulations. This has been proved by examining a variety of network models of the mentioned types. Especially tandem systems, where correlations will have the most inference have been considered.

Network models in which the priority discipline is only present at one node as well as network models with class changing are still under study.

*no class changes have been allowed.