Queueing Networks with Blocking
analysis, algorithms and properties

Simonetta Balsamo
Università Ca’ Foscari di Venezia
Dipartimento di Informatica
Venice, Italy

Queueing networks represent resource sharing and contention by a set of customers. Queueing networks with blocking consider resources with finite capacity queues and population constraints. Finite capacity of the queue is given by the number of customers in the service center and the network finite capacity. Blocking and deadlock are interdependent, with various blocking types: different behaviors of customer arrivals at a full node and of servers’ activity. Heterogeneous QNB: service centers may have different blocking types.

Outline
Queueing networks with blocking

I) Models of systems with finite capacity resources - population constraints
    • Types of blocking mechanisms
      • Various system behavior (network protocols, technologies)
    • Performance indices
      • Average (throughput, utilization, mean response time)
      • Distribution (queue length, blocking probability, effective throughput)
    • Analytical solution methods

II) Exact solution
    • Approximate solution methods
    • Solution algorithms, comparison, conditions

III) Some equivalence properties
    • Some application examples
    • Open research

Queueing networks with blocking (QNB): finite capacity queues

QNB analysis: exact and approximate methods, simulation
Various blocking types:
different behaviors of customer arrivals at a full node and of servers' activity.

Queueing networks with finite capacity queues:
- BAS: Blocking After Service
- BBS: Blocking Before Service
- RS: Repetitive Service Blocking

Queueing networks with (sub)network population constraint:
- STOP
- Recirculate

---

Types of blocking:
- BAS: Blocking After Service
- BBS: Blocking Before Service
- RS: Repetitive Service Blocking

If a job after its service attempts to enter a full node, is forced to wait in front of the sending server; the service is blocked until the job enters the destination node.

- Unblocking scheduling
- First Blocked First Unblocked

---

Blocking Types

QNB with finite capacity queues:
- BAS: Blocking After Service
- BBS: Blocking Before Service
- RS: Repetitive Service Blocking

Blocking After Service

- If a job after its service attempts to enter a full node, is forced to wait in front of the sending server; the service is blocked until the job enters the destination node.
- Unblocking scheduling
- First Blocked First Unblocked

---

Blocking Types

QNB with population constraints $n \in [L, U]$:
- $a(n) = 0$ for $n \in U$: load dependent arrival rate
- $d(n) = 0$ for $n < L$: load dependent service rate

**STOP blocking**
- If $d(n) = 0$ then service at each node is stopped
- Service is resumed upon a new arrival to the network.

**RECURSIVE Blocking**
- A job upon completion of its service at node $i$, leaves the network with probability $p_i(d(n))$, and it is forced to stay in the network with probability $p_i(1-d(n))$, where $p_i$ is the routing probability.
- That is, a job after its service at node $i$ is forced to stay in the network with probability $p_i(1-d(n))$ and is forced to enter a full node with probability $p_i d(n)$.
Deadlock

In queuing networks with finite capacity deadlock can occur with BAS, BBS, RS-FD. Prevention or detection and resolving techniques.

A simple prevention technique

| the overall network population | Total buffer capacity of the queues in each possible cycle of the network |

and for BAS and BBS $p_j = 0$ for each node $i$

NOTE

networks with finite capacity and RS-RD with irreducible routing matrix and the network population is less than the total buffer capacity do not deadlock.

QNB: performance indices

| Related to a single resource $i$ (a service center) |
| average indices |
| $U_i$ utilization |
| $k_i$ throughput |
| $L_i$ mean queue length |
| $T_i$ mean response time |

| Related to the overall network |
| average indices |
| $U$ utilization |
| $X$ throughput |
| $L$ mean population (for open networks) |
| $T$ mean response time |

deadlock $= 0$

customer passage times through the resource

distribution of $n_i$

mean passage time

job loss probability (for open networks)

Performance indices

For single server node $i$

| utilization $U_i = 1 - k_i(0) - PB_i$ |
| throughput $X_i = \mu_i \{ n_i(0) + PB_i(n_i) \} \mu_i(n_i) |
| $X_i = U_i \mu_i$ |
| mean queue length $L_i = \sum n_i n_i(n)$ |
| mean response time $T_i = L_i/\lambda_i |
| mean cycle time for node $i$ $T_i/\lambda_i |

$PB_i(n)$ probability that node $i$ is not empty and blocked when there are $n_i$ customers in $i$

$\forall$ customers in $i$

$PB$ definition depends on the blocking type

Effective utilization when the server is neither empty nor blocked

Effective throughput the useful work (for RS and BBS)

For the overall network

| overall blocking probability $PB = \sum_i PB_i(n_i)$ |

$PB$ definition depends on the blocking type
Analytical solutions for QNB

Evaluation of average performance indices and joint queue length distribution at arbitrary times (n)

- **exact** solution: based of Markov process analysis
  - Product-form solution of FCFS
  - Product-form solution of FCFS
  - Exact Product-form algorithms (on topology, blocking type,…)

- **approximate and bound** solution
  - Approximate analysis
  - Approximate analysis
  - Approximate analysis
  - Analytical solutions
  - Two-node cyclic network

A simple example: two-node cyclic network

FCFS service discipline
- **exponential service time**
- System state definition: $S = (S_1, S_2)$
  - RS or BBS blocking
  - RS server active, BBS server blocked
  - BAS blocking
  - where $s_i$ is the server state: $s_i = 1$ (active) $s_i = 0$ (blocked)

Birth-death Markov process
- Closed-form solution

Analyses and performance indices

Analytical solutions for QNB

Defining $S_i = (n_1, n_2)$ as system state:

- RS: $n_1 = 0$, $n_2 = B_1$
- BBS: $n_1 = B_1$, $n_2 = 0$
- BAS: $n_1 = n_2$

For infinite capacity queues (no blocking):

- $M = ((n_1, n_2), \Sigma)$
- $Q = (1/C)$

Markovian network

The network behavior can be represented by a homogeneous continuous time Markov process $M$

- Discrete state space
- Infinitesimal generator

Example:

- $P = (\rho, \rho)$
- $\rho = (n(S), S)\in E$

Solution of the global balance equations:

- $Q = 0$
- $\sum E n(S) = 1$
Analytical solutions for QNB: state definition

<table>
<thead>
<tr>
<th>State</th>
<th>( b_i = (n_i, d_i) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_i )</td>
<td>destination node of the next job that will exit from node ( i )</td>
</tr>
<tr>
<td>( b_i )</td>
<td>server state: active (1) blocked (0)</td>
</tr>
</tbody>
</table>

\( m_i = (m_{i_1}, m_{i_2}, \ldots, 0) \) is a \( i \)-th order Markov process

Queue of indices of the nodes blocked by node \( i \) if \( n_i \neq B \)

Unblocking scheduling

- RS: the server is always active
- BBS-SO: the server is blocked if \( n_i = 0 \) and \( n_j = B \)
- BBS-SNO: idem and \( n_i \neq B \)
- BBS-O: the server is blocked if at least one of the destination nodes of node \( i \) is full (full if \( p_j > 0 \) and \( n_j = B \))

Exact analysis of Markovian QNB

Solution algorithm for the evaluation of average performance indices and joint queue length distribution at arbitrary times \( t \) in Markovian QNB

1. Definition of system state and state space \( E \)
2. Definition of transition rate matrix \( Q \)
3. Solution of global balance equations to derive \( \pi \)
4. Computation from \( \pi \) of the average performance indices

This method becomes infeasible as \( |E| \) grows, i.e., proportionally to the dimension of the model (number of customers, nodes and chains)

→ exact product-form solution under special constraints
→ approximate solution methods

Analytical solutions for QNB: process definition

Different process transition rate matrices \( Q \) dependent on blocking type

\( Q = ||q(S, S')|| \)

- RS-RD

\( q(S, S') = \delta(n_i) \mu_i b(n_i) p_j \)

if \( S' = S + e \cdot e_i \)

\( q(S, S') = \lambda p_j b(n_i) \)

if \( S' = S + e_i \)

\( \lambda \) total arrival rate

\( b(n_i) \) blocking function of node \( i \)

\( \delta(n_i) = 0 \) if \( n_i = 0 \), \( \delta(n_i) = 1 \) otherwise, 1st iDM

\( e \) \( i \)-th position

\( q(S, S') = \sum_{S \in \text{possible states}} q(S, S') \)

Exact analysis of QNB: special cases

Subset of Markovian networks

Product-form solution of \( \pi \)

- Single class open or closed networks under certain constraints, depending on the network definition and the blocking type

\( \pi(S) = \frac{1}{G} \prod_{i=1}^{M} f_i \prod_{j=1}^{N} g_j \)

\( G \) normalizing constant

\( V \) total network population

\( V \) and \( g \) depend on network parameters \( (x, \mu) \) and population blocking type additional constraints

Various formulas F-I-F5 define functions \( V \) and \( g \) for different combinations of network topology blocking type

Computationally efficient exact solution algorithms

- Convolution Algorithm
- Mean Value Analysis
**Product-form heterogeneous QNB**

Formulas and admitted blocking types for each network topology, with additional constraints

<table>
<thead>
<tr>
<th>Network topology</th>
<th>Two nodes</th>
<th>Cyclic</th>
<th>Central server</th>
<th>Reversible routing</th>
<th>Arbitrary</th>
</tr>
</thead>
<tbody>
<tr>
<td>BAS</td>
<td>RS-RD</td>
<td>RS-FD</td>
<td>BBS-SO</td>
<td>RS-RD</td>
<td>RS-RD</td>
</tr>
<tr>
<td>F1</td>
<td>F2</td>
<td>F3</td>
<td>F5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Product-form formula**

For single class exponential networks with load dependent service rates \( \mu_i(n) = \mu_i \) (n)

- A-type nodes

- For single class exponential networks with load dependent service rates \( \mu_i(n) = \mu_i \) (n)

- State-dependent routing

- Blocking functions dependent on node and class

- A-type nodes

- Formula F3

- Multiclass central server networks with the class type of a job fixed in the system

- State-dependent routing depending on the class type

- Blocking functions dependent on node and class

- A-type nodes

**QNB Product-form: constraints and formulas**

Formula F3

Multiclass central server networks with the class type of a job fixed in the system

- State-dependent routing depending on the class type

- Blocking functions dependent on node and class

- A-type nodes

- Formula F3

- Multiclass central server networks with the class type of a job fixed in the system

- State-dependent routing depending on the class type

- Blocking functions dependent on node and class

- A-type nodes

**QNB Product-form: constraints and formulas**

Formula F3

Multiclass central server networks with the class type of a job fixed in the system

- State-dependent routing depending on the class type

- Blocking functions dependent on node and class

- A-type nodes

- Formula F3

- Multiclass central server networks with the class type of a job fixed in the system

- State-dependent routing depending on the class type

- Blocking functions dependent on node and class

- A-type nodes

**Exact analysis of QNB: product-form principles**

Most of the product-form solutions have been derived by applying reversibility of the underlying Markov process

Duality

Reversibility

The underlying Markov process of the QNB can be obtained by truncating the reversible Markov process of the network with infinite capacity (by the theorem on truncated Markov process): the same solution as the whole process normalized on the truncated sub-space holds. - product-form solution

Examples

- Two-node exponential single class cyclic networks

- Multiclass networks with BCPM, RS blocking and reversible routing P

- P is reversible

- Control variables \( \lambda_i, \rho_i \) for \( i \geq 1 \)
Exact analysis of QNB: product-form principles

**Duality**
- A dual network is obtained from the original one by reversing the connections between the nodes.
- Not-Empty-Condition of original network $\iff$ dual network without blocking.

**Examples**
- Exponential cyclic network with BBS or RS.
- Arbitrary topology networks with load-independent service rates for RS-RD blocking.
- Closed cyclic networks with phase-type (general) service distributions and BBS-SO blocking for which the throughput of the network is shown to be symmetric with respect to its population.

$$B = \sum B_i, \quad X(N-B) = X(B)$$

Product-form QNB: algorithms

**MVA**
- Direct computation of average performance indices.
- RS blocking, cyclic topology, load independent service rates.
- F2 product form solution.

**Duality**
- Based on the MVA algorithm for the dual network.
- Derivation of
  - Mean queue length $L_i$
  - Response time $\tau_i$
  - Throughput $X_i$
  - Utilization $U_i$
  - Mean busy period
  - Blocking probabilities

**Computational complexity** $O(MN)$

Exact analysis of QNB: special cases

**Symmetrical networks**
- Identical blocking type.
- Identical values of $\mu_i$ and $\eta_i$ for each node $i$.
- Routing $P$ where all rows are identical up to a rotation of the entries.
- Efficient computation of $\pi$ and average indices.

**Reduction algorithm**
- Based on exact aggregation of the Markov process, due to the special network structure.
  - Identification of a partition of $E$ in $K$ subsets $(E_k, 1 \leq k \leq K)$.
  - Decomposition-aggregation procedure.
  - For symmetrical networks: uniform conditional distribution.
  - Aggregated probabilities $\pi_k = \sum E_k \pi$.
  - With aggregated matrix $Q' = \sum E_k Q_k$, $Q_k[1] = 1$ for all $E_k$.

**Computation of $\pi_k$**
- Reduction to the computation of $\pi_k$: $O(K^3)$.
**Example of symmetrical networks**

Each node has the same probabilistic behavior

\[ p_i = \mu, B_i = B, 1 \text{stM} \]
\[ p_i \neq 0 \Rightarrow p_{i+1} = p_{i+2} = 0, 1 \text{stM} \]
\[ p_i \neq 0 \Rightarrow p_{i+1} = p_{i+2} = 0, 1 \text{stM} \]

where \( r = \frac{1}{K} \) if \( K \) is the outdegree of each node

Exponential service time, abstract service discipline (FCFS)

**Examples of symmetrical network topologies**

---

**Markov Process Decomposition**

Markov process with state space \( E \) and transition matrix \( Q \)

- Identify a partition of \( E \) into \( K \) subsets

\[ E = \bigcup_{k=1}^{K} E_k \]

- Decomposition of \( Q \)

- Decomposition-aggregation procedure

\[ \pi(S) = \text{Prob}(S | E_k) \]

\[ \pi^k \]

- Aggregated probabilities

- Computation of \( \pi(S) \) reduces to

- Computation of \( \text{Prob}(S | E_k) \) \( \forall S, \forall E_k \)

- Exact computation soon becomes computationally intractable

- Approximation of \( \text{Prob}(S | E_k) \) and \( \text{Prob}(E_k) \)

---

**Approximate analysis of QNB**

Many approximation methods

Most of them do not provide any bound on the introduced error

Validation by comparison with exact solution or simulation

**Basic principles**

- Decomposition applied to the Markov process or to the network
- Forced product-form solution
- Structural properties for special cases
- Maximum entropy

Various accuracy and time computational complexity

---

**Process and Network Decomposition**

Heuristics take into account

- The network model characteristics
- The blocking type

NOTE: the identification of an appropriate state space partition affects

- The algorithm accuracy
- The time computational complexity

If the partition of \( E \) corresponds to a network partition into subnetworks, network decomposition subsystems are (possibly modified) subnetworks

The decomposition principle applied to QNB is based on the aggregation theorem for QNB

1. **Network decomposition** into a set of subnetworks
2. Analysis of each subnetwork in isolation to define an aggregate component
3. Definition and analysis of the new aggregated network
Network Decomposition

1. network decomposition
   NP-complete problem: critical issue

2. analysis of isolated subnetworks
   choose simple subnetworks
   apply efficient solution methods

3. aggregated network analysis
   aggregation theorem: exact only for product-form networks
   approximation otherwise
   unknown error

Various approaches determine the subnetwork parameters

Approximate methods for QNB

Method comparison
- model assumptions
- algorithm rationale
  constraints on the network parameters
  topology, service distribution, blocking type
- performance comparison
  accuracy
  efficiency
  class of models to which they can be applied
  - model parameters
    nodes, customers, topology, service rates, queue capacity
  - symmetry of network parameters

- Six significant approximate methods for closed QNB
- Four significant approximate methods for open QNB

Experiments

Network Decomposition

- approximations based on the forced application of the exact aggregation technique for product-form QN without blocking
- low computational cost
- accuracy: experimental results suitable for many practical cases
  BUT the approximation error is UNKNOWN

- many approximations are based on iterative solution of subsystems or subnetworks
  Iterative aggregation-disaggregation
  speed and proof of convergence

- few approximate techniques with known accuracy
  bound solutions can be used as approximation methods with known accuracy

- open issue: solution of general classes of heterogeneous QNB

Approximate methods for closed QNB

<table>
<thead>
<tr>
<th>Method</th>
<th>Network Constraints</th>
<th>Blocking Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Throughput Approximation</td>
<td>cyclic G/M/1/B</td>
<td>BAS or BBS-SO</td>
</tr>
<tr>
<td>Network Decomposition</td>
<td>cyclic G/M/1/B</td>
<td>BBS-SO</td>
</tr>
<tr>
<td>Variable Queue Capacity</td>
<td>cyclic G/M/1/B</td>
<td>BBS-SO</td>
</tr>
<tr>
<td>Matching State Space</td>
<td>general G/M/1/B</td>
<td>BAS</td>
</tr>
<tr>
<td>Approximate MVA</td>
<td>general G/GE/1/B</td>
<td>BS-RD</td>
</tr>
<tr>
<td>Maximum Entropy Algorithm</td>
<td>general G/GE/1/B</td>
<td>BS-RD</td>
</tr>
</tbody>
</table>

M exponential, G general, GE generalized exponential
A/Bi Kendall’s notation:
- A: customer interarrival time distribution
- B: service time distribution
- i: number of identical servers
Observations

Cyclic Networks
Exponential service times
Performance index: network throughput as a function of the network population: X(N)

<table>
<thead>
<tr>
<th>Method</th>
<th>Blocking Type</th>
<th>Key idea</th>
</tr>
</thead>
<tbody>
<tr>
<td>Throughput Approx. [Chung-Perros 90]</td>
<td>RSS-RR</td>
<td>Exact model analysis for some network population. Throughput interpolation by varying network population.</td>
</tr>
<tr>
<td>Network Decomp. [Dallery-Frein 93]</td>
<td>RSS-RR</td>
<td>Network decomposition into single nodes analyzed in isolation as M/M/1/B queues</td>
</tr>
<tr>
<td>Variable Queue Capacity Decomp. [Sun-Dahl 96]</td>
<td>RSS-RR</td>
<td>Network aggregation of the set of finite capacity queue nodes in a single composite node having state dependent service rate and variable buffer size</td>
</tr>
</tbody>
</table>

Algorithm for closed QNB: comparison

<table>
<thead>
<tr>
<th>Method</th>
<th>Performance Indices</th>
<th>Accuracy</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Throughput Approx.</td>
<td>X(N)</td>
<td>Very good</td>
<td>Poor for networks with more than B nodes</td>
</tr>
<tr>
<td>Network Decomp.</td>
<td>X(N)</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>Variable Queue Capacity Decomp.</td>
<td>X(N)</td>
<td>Valid</td>
<td>Valid</td>
</tr>
<tr>
<td>MVA</td>
<td>X(N)</td>
<td>Valid</td>
<td>Good</td>
</tr>
<tr>
<td>Approximate MVA</td>
<td>X(N)</td>
<td>Valid</td>
<td>Valid</td>
</tr>
<tr>
<td>Maximum Entropy Algorithm</td>
<td>X(N)</td>
<td>Valid</td>
<td>Poor for other performance indices</td>
</tr>
</tbody>
</table>

Approximate methods for open QNB

<table>
<thead>
<tr>
<th>Method</th>
<th>Network Constraints</th>
<th>Topology</th>
<th>Node type</th>
<th>Blocking Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tandem Type Network Decomposition [Dallery-Frein 93]</td>
<td>tandem</td>
<td>G/M/1/B</td>
<td>BAS</td>
<td></td>
</tr>
<tr>
<td>Acyclic Network Decomposition [Lee-Altick 93]</td>
<td>acyclic</td>
<td>G/M/1/B</td>
<td>BAS</td>
<td></td>
</tr>
<tr>
<td>Maximum Entropy Algorithm [Kouvatsos-Xenios 89]</td>
<td>general</td>
<td>G/G/E/1/B</td>
<td>RS-RD</td>
<td></td>
</tr>
</tbody>
</table>

MSS and AMVA assume networks with exponential service time and evaluate the network throughput.
ME Algorithm assumes generalized exponential service time and evaluates the queue length distribution and average performance indices.

Methods

The approximation principle is network decomposition for all the algorithms. One-node subnetworks as M/G/1/B queue by Tandem Phase-Type Decomposition
M/M/1/B queue by other algorithms
Last algorithm applies the maximum entropy principle
Algorithm for open QNB: comparison

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tandem Exponential</td>
<td>Very good</td>
<td>Very good</td>
</tr>
<tr>
<td>Network Decomposition</td>
<td>for all performance indices</td>
<td></td>
</tr>
<tr>
<td>Tandem Phase-Type</td>
<td>Very good</td>
<td>Slow when</td>
</tr>
<tr>
<td>Network Decomposition</td>
<td>for all performance indices</td>
<td>applied to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>networks where</td>
</tr>
<tr>
<td></td>
<td></td>
<td>all the nodes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>have finite</td>
</tr>
<tr>
<td></td>
<td></td>
<td>capacity and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fail otherwise</td>
</tr>
<tr>
<td>Acyclic Decomposition</td>
<td>Very good</td>
<td>Very good</td>
</tr>
<tr>
<td></td>
<td>for all performance indices</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum Entropy</td>
<td>Good for all</td>
<td>Fair</td>
</tr>
<tr>
<td>Algorithm</td>
<td>performance indices</td>
<td></td>
</tr>
</tbody>
</table>

All algorithms evaluate for each node i:

- \( n_i \): queue length distribution
- \( \lambda_i \): node throughput
- \( \mu_i \): node mean response time

Observations

QNB: equivalence Properties

- equivalences in terms of
  - state probability distribution \( \pi \)
  - average performance indices
  - passage time distribution

Most of the equivalences derive from the identity of the network processes \( \pi \).

Remark: Even if two networks have identical Markov processes, the meaning of corresponding states may be different.

\( \pi \) performance measures may be NOT equivalent.

\( \pi \) equivalence in terms of \( \pi \) does NOT necessarily lead to equivalence in terms of average performance indices.

\( \pi \) equivalence between networks

- with and without blocking
- with different blocking types
- homogeneous and non-homogeneous networks

\( \pi \) extension of efficient computational algorithms (MVA and Convolution) and solution methods to QNB

(e.g.: aggregation technique)
null
Application example of QNB

Store-and-forward packet switching networks

Circuit switching networks
- data packets
- travel through the network or wait to be transmitted
- routing
- system resources
- shared by the data to be transmitted
- network topology
- allocation of link capacity for the connection (circuit switching)

Problems
- Buffer allocation
  - Determine the amount of buffer space to be allocated to each station to optimize system performance (e.g., maximize network throughput, minimize end-to-end delay)
- Routing algorithm
  - Scheduling

Performance measures
- Average packed delay over the entire network
- End-to-end delay for pairs source-destination
- Buffer occupancy
- Loss probability

Example of heterogeneous QNB: computer-communication system

Computer System

Communication Subnetwork

C1, C2: computer CPU subsystem
D1, D2: computer Disk subsystem
N1, N3: computer network access
N2, N4: communication links

Customers represent jobs (in Computer Systems) and packets (in Communication Subnetwork)

Under exponential assumption:
- heterogeneous QNB reducible to homogeneous QNB RS-RD
- if D1, D2 have RS-RD blocking, product-form solution F2 - convolution algorithm

Conclusions and open research

Queueing Network models with finite capacity queues and blocking can model systems with finite capacity resources and population constraints

QNB are difficult to analyze
- Various exact and approximate algorithms
  - Markov process analysis
  - various approximation with different efficiency, accuracy, model constraints and parameters

Heterogeneous networks
- equivalence and reducibility properties

Open problems
- algorithms for general heterogeneous QNB
- efficient solution tools
Questions?
For further information

S. Balsamo, D. Kouveliotou Special Issue "Queueing Networks with Blocking." Performance Evaluation, 2003, 51-12

References

Books

Special Issues


Surveys papers


Papers

References

References

4


References

5


M. Nak, M. F. "Two queues in series with a finite intermediate waiting room" J. Appl. Prob. 5 (1968) 123-140.


Additional method information

Convolution algorithm for product-form closed QNB

Details of the approximation algorithms for closed QNB open QNB

Observations on method comparison

Details on product-form conditions