CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
Telephone 31 - 20 592 9333, telex 12571 (mactr nl),
telefax 31 - 20 592 4199

CWI is the nationally funded Dutch institute for research in Mathematics and Computer Science.

Performance evaluation of parallel and distributed systems
Solution methods

Proceedings of the third QMIPS workshop
Part 1

O.J. Boxma, G.M. Koole (eds.)

1991 Mathematics Subject Classification: 60K25, 68M20.
ISBN 90 6196 444 x
NUGI-code: 811

Printed in the Netherlands

UNE IMP

EHE ★ ★ ★ CWI

★ ★

★QMIPS★

★ ★

INR ★ ★ ★ UER

ZAR UTO

Quantitative
Modeling
In
Parallel
Systems

The QMIPS project is a collaborative research project supported by the CEC as ESPRIT-BRA project no 7269. It is being carried out by the following organisations: CWI (Amsterdam), EHEI (University of Paris V), Imperial College (London), INRIA (Sophia-Antipolis), University of Erlangen, University of Newcastle, University of Torino and University of Zaragoza.

Contents
Part 1

Part 2

Introduction

These are the proceedings of the third QMIPS workshop, held in Torino, Italy, on September 25 and 26, 1993. The QMIPS project is a collaborative research project supported by the European Union, and it is carried out by 8 organizations from 6 different European countries. It is concerned with quantitative modeling in parallel and distributed systems. Within the framework of the QMIPS project several workshops are being organized. After workshops in Sophia-Antipolis (France) on Petri nets, and in Erlangen (Germany) on modeling formalisms, this workshop focused on solution methods.

Three steps can be distinguished in the analysis of parallel or distributed systems. The first is modeling, using one of the available formalisms. Depending on the formalism used, a solution method is employed to obtain performance measures for the system. This second step is the subject of these proceedings. The third step is the optimisation of the system. Research in this area is presented at the fourth QMIPS workshop in London, on April 14 and 15, 1994.

The proceedings start with two survey papers, one on solution methods for queueing models, and one on solution methods for Petri net models. The other 16 papers, all concerned with current research topics, are divided in three parts, depending on the formalism used: queueing, Petri nets or the (Max,+) algebra.

The first formalism is queueing. The paper by ETTL AND MITRANI analyses two queueing models using the recently developed spectral expansion method. BOXMA AND VAN HOUTUM apply the compensation approach to a 2×2 switch, and MITRANI AND WRIGHT solve a two-dimensional queueing problem using the boundary value technique. The next two papers deal with queueing models with negative customers, which are customers with the ability to cancel regular customers. The paper by FOURNEAU, GELENBE AND SUROS extends product form results for regular queueing networks to networks with negative customers. The paper by HARRISON AND PITEL studies tandem models which do not have product form solutions, and analyses them using the boundary value technique. KOOLE shows that the power series algorithm, which has been applied to many queueing models, can also be used for general Markov chains.

The part on Petri nets starts with the paper by BOUCHERIE AND SERENO. It characterises product form Petri nets in terms of the structure of the net. The paper by SERENO AND BALBO considers product forms as well, but focuses on computational algorithms. Also the paper by CHIOLA, ANGLANO, CAMPOS, COLOM AND SILVA studies a technique originating from queueing, namely operational analysis, which leads them to performance bounds. FRANCESCHINIS AND MUNTZ derive performance bounds for certain Petri nets that exhibit symmetry. CAMPOS, COLÓM, SILVA AND TERUEL study a model consisting of several sequential processes communicating through buffers, and derive both qualitative and quantitative results. The paper by BACCELLI AND GAUJAL is concerned with free choice Petri nets. For this class of nets qualitative properties are derived. In the paper by CAMPOS, COLOM, JUNGNITZ AND SILVA marked graphs (Petri nets where each place has only one input and output arc) are considered, and a technique is introduced to approximate the throughput of marked graphs. CANALES AND GAUJAL also study marked graphs, and exhibit the inherent parallelism to derive efficient parallel simulation procedures.

Marked graphs are strongly related to the (Max,+) algebra, and as such the two papers on this algebra are relevant to the study of Petri nets. JEAN-MARIE studies stochastic event graphs by an analysis based on the (Max,+) algebra. MAIRESSE derives some deep results on small matrices in the (Max,+) algebra.

Finally, a few words of thanks. We thank the local organizer G. Balbo and his co-workers (University of Torino) for taking care of the local arrangements and for selecting such a wonderful location for the workshop. We should like to express our gratitude to the Centre for Mathematics and Computer Science (CWI) for its support in publishing these proceedings. We are in particular grateful to Yvonne Samseer from the CWI typesetting department for her many valuable contributions to the preparation of the final manuscript, and to managing editor Wim Aspers.

Onno Boxma and Ger Koole

Queueing-Theoretic Solution Methods for Models of Parallel and Distributed Systems*

Onno Boxma[†], Ger Koole[†] & Zhen Liu[‡]
[†] CWI, Amsterdam, The Netherlands
[‡] INRIA-Sophia Antipolis, France

This paper aims to give an overview of solution methods for the performance analysis of parallel and distributed systems. After a brief review of some important general solution methods, we discuss key models of parallel and distributed systems, and optimization issues, from the viewpoint of solution methodology.

1 INTRODUCTION

The purpose of this paper is to present a survey of queueing theoretic methods for the quantitative modeling and analysis of parallel and distributed systems. We discuss a number of queueing models that can be viewed as key models for the performance analysis and optimization of parallel and distributed systems. Most of these models are very simple, but display an essential feature of distributed processing. In their simplest form they allow an exact analysis. We explore the possibilities and limitations of existing solution methods for these key models, with the purpose of obtaining insight into the potential of these solution methods for more realistic complex quantitative models.

As far as references is concerned, we have restricted ourselves in the text mainly to key references that make a methodological contribution, and to surveys that give the reader further access to the literature; we apologize for any inadvertent omissions. The reader is referred to Gelenbe's book [65] for a general introduction to the area of multiprocessor performance modeling and analysis.

Stochastic Petri nets provide another formalism for modeling and performance analysis of discrete event systems. The reader is referred to the survey paper of Murata [127] for results of their qualitative analysis. The use of this tool for performance evaluation of parallel and distributed systems has recently become popular, as is illustrated in the special issue of *J. of Parallel and Distributed Computing* (Vol. 15, No. 3, July 1992). A survey on recent results of their quantitative analysis is found in the paper of Baccelli et al. [6] in these proceedings.

---

*Supported by the European Grant BRA-QMIPS of CEC DG XIII.

Simulation is an important method for solving queueing models. In this paper, we will not discuss that approach. The interested reader is referred to the books of Mitrani [125], Sauer and MacNair [146] and Rubinstein [145].

The paper is organized as follows. Section 2 contains a global discussion of some solution methods that have been successful in the performance analysis of parallel and distributed systems. Six key models for this performance analysis are discussed in Section 3, with an emphasis on solution methodology. Section 4 is concerned with mathematical techniques for the optimal control of distributed systems. We distinguish between load balancing, routing, server allocation and scheduling.

2 SOLUTION METHODS

The publication of J.W. Cohen's 'The Single Server Queue' [37] in 1969 marked the end of an era in queueing theory, in which the emphasis in queueing research had been placed on the exact analysis of models with one server and/or one queue. Around 1970 successful applications of queueing theory to problems of computer performance began to appear. Rather simple queueing network models turned out to be able to yield quite accurate predictions of the behaviour of complex computer systems, thus stimulating queueing network research. Extensive queueing network results have been obtained in the seventies and eighties and have been made available for computer engineering purposes by the introduction of efficient numerical algorithms (see the surveys of Kleinrock [100] and Lavenberg [109]).

The performance analysis of parallel and distributed systems leads in a natural way to multidimensional queueing models. Generalization of the single-queue solution methods to those models is straightforward only in rare instances; and adaptation of the queueing network results to parallel and distributed systems is usually only possible by making gross simplifications. In this section we discuss a few solution methods that try to bridge this gap, and that have been successful in analyzing the performance of a number of (often admittedly simple) models of parallel and distributed systems: Product-form solutions, some methods from complex function theory, a number of analytic-algorithmic methods, heavy and light traffic approximations, the large deviation technique, and state recursions. The choice of these methods, above, e.g., aggregation and decomposition methods is undoubtedly influenced by the research interests of the authors. The field of performance analysis of parallel and distributed systems is still in its infancy, and it is yet far from clear which methods have the biggest potential for capturing the characteristic features of parallel systems.

2.1 Product-form solutions

An important contribution of queueing network theory is that, under certain assumptions, it allows one to obtain a simple exact solution for the joint queue length distribution in a separable form: the *product form* [96, 163, 158]. The reader is also referred to Liu & Nain [114] for recent extensions, efficient computational algorithms, and sensitivity analysis of product form queueing networks, and to Disney & König [50] for an extensive survey of queueing networks and their random processes.

Although the modeling of parallel and distributed systems only seldom leads to product forms, queueing networks do provide an important and widely used tool for modeling parallel and distributed systems. We mention two interesting product-form applications. Heidelberger & Trivedi [83] present a class of parallel processing systems in which jobs subdivide in several asynchronous tasks; they approximate this non-product-form network iteratively by a sequence of product-form networks. An interesting new development in product-form theory has been stimulated by the performance analysis of resource request and allocation models with positive and negative signals; they were shown to give rise to product-form networks with positive and negative customers [66, 68]. The latter results are surveyed in [67]. See Harrison & Pitel [81] for some recent results and further references on *sojourn time* distributions in networks with positive and negative customers.

2.2 Methods from complex-function theory

The modeling of queueing systems with multiple queues and/or multiple servers frequently leads to multi-dimensional models with multiple unbounded components; in particular to the analysis of Markov processes whose state space is the N-dimensional set of lattice points with integer-valued non-negative coordinates. The functional equations arising in the analysis of such processes (obtained after taking transforms of, say, the joint queue length distribution) usually present formidable analytic difficulties.

For the two-dimensional case, however, techniques have been developed which often make it possible to reduce these functional equations to standard problems of the theory of boundary value equations (Wiener-Hopf, Dirichlet, Riemann, Riemann-Hilbert) and singular integral equations. Pioneering papers are those of Eisenberg [55] (transforming a two-queue polling problem into a Fredholm integral equation) and Fayolle & Iasnogorodski [57] (transforming a problem concerning two processors with coupled speeds into a Riemann-Hilbert boundary value problem). A systematic and detailed study of the 'boundary value method' is presented by Cohen & Boxma [43], with applications to various queueing problems: a two-queue polling problem, the shorter queue model, processors with coupled speeds, the M/G/2 queue. A concise exposition of the method, and several applications and references, are presented in [39]. A detailed investigation of random walks on the two-dimensional lattice in the first quadrant is continued by Cohen in [40]. This has led to a better understanding of, among others, the ergodicity conditions and the usefulness of the concept of (boundary) hitting points.

2.3 Analytic-algorithmic methods

The methods mentioned in the previous subsection are mainly applicable to some specific 'two-dimensional' models, and even then the performance measures are not always directly available. Therefore various analytic-algorithmic methods have been developed to solve multi-dimensional queueing systems. For multi-dimensional models with all but one component finite there are good analytic-algorithmic methods, like the well known *matrix geometric method* (Neuts [135]) and the related spectral method of Mitra & Mitrani [126]; see also the interesting and extensive methodological discussion by Gail et al. [64]. Below we discuss the power series algorithm and the compensation approach, two methods that are not yet so well known, that are mathematically interesting and that exploit the stochastic properties of queueing systems more than general methods based on, say, state space truncation and solving large systems of equations for Markov chains. It should be observed, though, that recently much progress has been made in numerically handling large Markov chains. Several approaches are presented in the conference proceedings [150]; see also the survey of Grassmann [73].

The power series algorithm (introduced by Hooghiemstra et al. [85]) is a numerical procedure which can, formally, be applied to any Markov process (cf. [107]). It writes the stationary distribution of the process as a power series of some parameter, in queueing applications usually the load of the system. In a series of papers Blanc (with co-authors) has shown that the algorithm works well for many multi-dimensional queueing systems (see his survey [20]). However, the convergence properties and error estimates of the algorithm are still unknown, and therefore no guarantee can be given as to the convergence of the method for arbitrary models.

The compensation approach (developed in Adan's PhD thesis [1]) can be applied to two-dimensional homogeneous random walks in the first quadrant without transitions to the north, north-east and east. It writes the stationary distribution as a sum of product forms, which all satisfy the steady state equations on the interior, and where each additional term ensures, alternatively, that the steady state equations on the horizontal, respectively vertical, axis are satisfied. Generally the algorithm can be shown to converge exponentially fast. The algorithm is developed by Adan, Wessels & Zijm [2] for the shortest queue model. Other queueing models which are studied are a multiprogramming queue ([1], ch. 4), and the 2×2 clocked buffered switch of an interconnection network [26]. For the latter model, the method has been extended to a 3-dimensional case [159]. [26] indicates a link between the compensation approach and the boundary value method mentioned in subsection 2.2. Recent work of Cohen [41, 42] considerably adds to this insight. In [41] he studies the class of two-dimensional nearest neighbour random walks without transitions to the north, north-east and east, that is also considered in [1]. He shows that the bivariate generating function of the stationary distribution can be represented by a meromorphic function—an analytic function apart from a finite number of poles in every finite domain. The poles appear in fact as powers in the product forms in Adan's solution representation (cf. the correspondence between the representations $1/(1-az)$ and $\sum a^n z^n$). Cohen exposes the construction of this representation as a meromorphic function. He does this in much more detail for a special case of this class of random walks, the symmetrical shortest queue: he shows [42] how all poles, *and* all zeros, of the meromorphic generating function can be determined from the original functional equation. This leads to a simple expression for the main performance measures, which are easily calculated with any desired accuracy.

2.4 Heavy and light traffic approximations

Approximation techniques present an alternative approach to numerical methods for solving analytically intractable queueing systems. Heavy and light traffic approximations are among the most popular techniques of this kind.

By *heavy traffic* we mean that the system approaches saturation, so that the queues are nonempty most of the time. In this case, the queue lengths, when properly normalized, can be approximated by Brownian motions with drift, which leads to a diffusion approximation of the system. The reader is referred to the survey of Glynn [72]. For generalized Jackson queueing networks, Harrison & Williams [78] prove the existence of stationary distributions of diffusions and their product form. They also show [79] that the notion of quasireversibility for queueing networks extends to the Brownian limit. Closed queueing networks are analyzed in Harrison, Williams & Chen [80]. Not all multiclass queueing networks can have such approximations (see e.g. [45]). Sufficient conditions for the existence and uniqueness of Brownian models are established in Reiman et al. [140] and Dai & Williams [46].

In *light traffic* approximations, a performance measure is considered as a function of the arrival rate. Derivatives of this function are computed at point zero. The light traffic approximations are developed in Burman & Smith [27, 28] for a single queue and in Reiman & Simon [139] for an open queueing network.

Approximations for moderate traffic can be obtained by interpolating heavy and light traffic approximations.

2.5 Large deviations

Although there is an extensive literature of large deviations on Markov processes (see Deuschel & Stroock [49], Dembo & Zeitouni [48]), it is only recently that these techniques have become important tools for solving queueing systems. Large deviations principle has been proved, under different statistical assumptions, for single queues, see e.g. Chang [31], Duffield & O'Connell [52], Liu et al. [116], and for queueing networks, see e.g. Tsoucas [157], Dupuis & Ellis [53].

A comprehensive treatment is provided in the forthcoming book [54]. Large deviations estimates have also been applied to rare event simulations, see Chang et al. [33].

2.6 State recursion

For queueing systems with synchronization constraints, as frequently occur in parallel systems, one can often write the dynamics in the form of a state recursion equation, generalizing Lindley's equation: Baccelli & Makowski [13], Baccelli & Liu [11]. Some of these state recursions lead to solvable integral equations as in the recent work of Jean-Marie [91]. Among the other techniques which were proposed based on state recursions, we would quote

- bounds: computable bounds on noncomputable stochastic models can often be obtained using stochastic ordering techniques (e.g. convex ordering, Schur convexity, association etc.). A good reference for this is the book by D. Stoyan [151]. A survey on the application of these techniques to synchronization problems can be found in [13].
- large deviation estimates as in Baccelli & Konstantopoulos [9]. This technique is based on the computation of the Cramer Legendre transform of the Perron Frobenius eigenvalue of the Laplace transform of the matrix that shows up in the state recursion.

3 KEY MODELS

In this section we discuss some queueing models that can be viewed as key models for the performance analysis of parallel and distributed systems: fork-join, task graph, resequencing, shortest queue, polling and time warp models. The discussion is methodologically oriented, which has also guided our choice of references.

3.1 Fork-Join Model

The Fork-Join model is a simple queueing model of a parallel processing system. It consists of c parallel processors, each with a local queue. Each arriving job consists of c tasks, who each join the queue of a different processor (the fork primitive). A job is completed if all its tasks have completed service (the join primitive). Thus the model consists of c interrelated parallel queues (which are stochastically dependent due to the simultaneous arrivals).

For $c = 2$ and Poisson arrivals this model has been studied analytically. Flatto & Hahn [58] solve the model for inhomogeneous exponential servers, and obtain the limiting distribution as the number of tasks in one of the queues grows to infinity. This result is generalized by Wright [170], who also allows jobs consisting of a single task to join the system. Baccelli [5] solves the model for general, but exchangeable, service times using complex-function theory methods. De Klein [99] solves the model with general service times using the boundary value approach.

A completely different approach to obtain the asymptotic results is used by Shwartz & Weiss [148] (see also the afterword in [170]), who use large deviations and reversibility. Given there are n tasks in the second queue, they use reversibility to show that this queue built up with arrival and service rate reversed. It follows from the theory of large deviations that the moment at which the rates are reversed is almost deterministic, for n large. As the arrival rates of both queues are the same, conclusions can be drawn for the arrival rate in the first queue, and using transient results for the $M|M|1$ queue, the asymptotics are derived.

Because the analytic results only hold for $c = 2$ (and even then, performance measures are hard to obtain), attention has been paid to approximations and bounds. Both Nelson & Tantawi [130, 131] and Baccelli et al. [14] derive bounds for the system. In [131] bounds on the mean job response time are derived using inequalities on the maximum of associated random variables. In [14] the exponential conditions are dropped, and bounds on various performance measures are derived, again using associated random variables, but also stochastic orderings. It is interesting to note that both papers show that the response time grows logarithmically in the number of processors. Varma & Makowski [162] present an approximation for symmetric fork-join queues, interpolating between light-traffic and heavy-traffic results.

Kim & Agrawala [97] provide an algorithm to obtain the response times in the case of Erlang service time distributions.

Nelson et al. [133] compare the Fork-Join model with three other models with and without local queues and distributed processing. A central queue and distributed processing (which is equivalent to an $M|M|c$ queue with batch arrivals) performs best.

3.2 Task Graph Models

Directed acyclic graphs are frequently used to represent parallel programs, and are referred to as *task graphs*. In a monoprogramming system (i.e. a system where at most one parallel program runs at any time), the computation of program completion time can be performed by PERT techniques. We refer the reader to Baccelli et al. [8] for a survey on these techniques.

In order to model multiprogramming systems (i.e. systems where more than one parallel program can run simultaneously), queueing models with extended (synchronization) primitives can be used.

The Fork-Join model can be considered as the ancestor of a line of models involving the execution of task graphs on parallel machines. Acyclic Fork-Join queueing models (corresponding to acyclic task graphs with precedence constraints) were introduced by Baccelli, Massey & Towsley in [15], where also a relation is indicated with the resequencing model that is considered below. The basic Fork-Join model is a special case of this acyclic case when all tasks have a single predecessor and a single successor.

A more general model with identical task graphs statically mapped on a set of processors was then introduced by Baccelli & Liu in [11]. The acyclic Fork-Join model mentioned above is a special case of this one when the number of processors is equal to the number of tasks in the graph. The model in [11] involves a non-trivial stability condition (which was recently understood in terms of so-called $(\max, +)$ Lyapunov exponents) and integral equations for the response times of tasks on processors that can be seen as the plain generalization of Lindley's integral equation. However, exact solutions of these equations are difficult except for very special cases.

Several variations on this basic model were proposed by the same authors in relation with distributed data base models ([10] and [113]). The main results for these models bear on (i) the shape of the stability region, (ii) the computation of the throughput either by stochastic ordering or using large deviation estimates as in [9], and (iii) bounds based on stochastic ordering. These models were also investigated using various asymptotic limits including the light traffic limits in the work of Varma [161] and diffusion limits as in the Stanford school around M. Harrison and in particular the work of Nguyen [136, 137].

3.3 Resequencing

The first stochastic resequencing models were infinite-server models, proposed in the context of reordering of packets in data communication networks. Kamoun, Kleinrock & Muntz [95] studied the exponential service time case using differential equations, and Baccelli, Gelenbe & Plateau [7] studied the non-exponential case using Wiener Hopf factorization. The model is basic in serialisation problems which arise quite naturally in various distributed algorithms. Models with no queueing effects were also considered by Harrus & Plateau [82] and by Varma [160], using analytical techniques. Recently Downey [51] made interesting new connections between the model considered in [7] and the cost of synchronization in parallel systems. The techniques used in these papers are mainly analytical ones based on complex variables, allowing to get simple series representations for the moments of response times etc.

In [166] Whitt presents a general exploration of overtaking phenomena in queueing networks; this includes an investigation of disordering in multiserver queues. He analyzes the number of jobs overtaken by an arbitrary job for $GI/M/s$ and $M/GI/s$ models with the First Come First Serve (FCFS) service policy. Iliadis & Lien [88] explicitly calculate the resequencing delay for two heterogeneous servers under two different threshold-type scheduling disciplines.

Other lines of thought consist in looking at

- more elaborate serialisation algorithms (e.g., timestamp ordering or two phase locking [16], [13]). The analysis method is essentially that of state recursions.
- more structured disordering structures like interconnection networks as considered in the thesis of A. Jean-Marie [89] (see also [90]), where complex-analysis methods are used to compute the moments of the resequencing delays.

Resequencing is surveyed in [13].

3.4 The shortest queue and the smallest workload model

An example of a model with distributed processing is the *shortest queue model.* Here arriving customers join out of two (or more) queues the one with the least customers in it. Usually the arrival process is taken to be Poisson and the service times are taken exponential. The idea behind joining the shorter queue is that it *balances* the load in the system. For qualitative questions concerning this model, see section 4.2. Here we will deal with the quantitative aspects, i.e. with the *performance analysis* of the 2-queue shortest queue model.

Complex-variable methods have led to an exact analysis of the joint queue length process. Kingman [98] and Flatto & McKean [59] use a uniformization technique to determine the equilibrium distribution in the case of equal service rates. Fayolle and Iasnogorodski in their theses [56, 86] show that, even for asymmetric service rates, the problem can be reduced to a — generalized — Riemann-Hilbert boundary value problem; see also [43]. Knessl et al. [103] develop a scheme to obtain approximations for the joint queue length distribution, valid when one of the queue lengths is large. Foschini & Salz [60] employ a heavy traffic diffusion approximation. Interesting numerical approaches are proposed by Adan et al. [2] (the compensation approach), Blanc [19] (the power series algorithm, applicable to the case of more than two queues and general service times), Gertsbakh [71] (the matrix-geometric method) and Zhao & Grassmann [173] (who present an algorithm based on the results of [59]). Halfin [77] employs linear programming techniques to obtain bounds, and Nelson & Philips [134] present mean response time approximations for the case of K queues and general interarrival and service time distributions, assuming in their approximation method that the various queue lengths can differ by at most one.

For the 2-queue model where the customers are assigned to the queue with the smallest workload (and general service times) a performance analysis has been presented in [104]. Formal asymptotic approximations are constructed for the two-dimensional workload process, treating separately the asymptotic limits of heavy traffic, light traffic and large buffer contents. Cohen [36] presents an exact analysis of this M/G/2 queue, using a Wiener-Hopf decomposition. Cohen [38] also solves the 2-queue model with server priority for the longer queue (in a sense dual to the shortest queue model); here he uses a translation into a Riemann boundary value problem of a type that was not studied earlier in a queueing context.

3.5 Polling

The performance analysis of distributed systems often gives rise to single-server multi-queue *polling* models. The characteristic feature of polling models is that the server is moving between queues (which possibly requires switchover times), implying that the priority of the queues is dynamically (e.g., cyclically) changing. Some examples are token passing schemes in local area networks with distributed channel access control, and resource arbitration and load sharing in multiprocessor computers. Many computer-communication examples of polling can be found in [74, 112, 153].

In a single-server cyclic polling model, the joint queue length process can — under some conditions on the service disciplines at the queues — be represented by a multi-type branching process with immigration [141]. The theory of such branching processes then immediately yields necessary and sufficient ergodicity conditions, and a complete solution for the joint queue length distribution. Unfortunately, the branching property does not hold for several important service disciplines, like those that put a limit on the number of services or the time of a server visit. In exceptional two-queue cases of the latter class, the joint queue length distribution can be determined by using the theory of Riemann-Hilbert boundary value problems [25, 43].

Proving ergodicity conditions for polling models generally is a challenging mathematical problem, for which recently considerable progress has been made; cf. the approach of [70] (based on stochastic dominance techniques and the well-known Loynes stability criteria for a queue in isolation), [4] (which uses Lyapunov functions for the verification of Foster's criterion), and [63] (based on a stochastic monotonicity property of the multidimensional queue length Markov chain at polling instants).

A quite generally valid result for (even non-cyclic) polling models is the pseudo-conservation law—an exact expression for a weighted sum of the mean queue lengths or mean waiting times [23]. The pseudo-conservation law has been extensively used to develop mean waiting time approximations.

Leung [110] has developed an interesting numerical procedure, based on the fast Fourier transform, that enables one in principle to determine polling performance measures with any required accuracy. The power series algorithm [18] is also applicable to a large class of polling models. An essential difficulty of these numerical techniques is their large computational complexity.

Takagi [152] gives an extensive bibliography of polling studies.

3.6 Time Warp

Simulations are usually well suited for parallel processing, especially if the physical model to be simulated consists of several components which can be simulated on different processors. Messages sent between the processors deal with the interaction between the components. A method to synchronize the components is the *Time Warp* protocol, as introduced by Jefferson [94]. Each processor continues the simulation, handling the already arrived messages. If a message arrives which should have been handled before, the processing *rolls back* to a point in time before the time associated with the message, and execution starts again. This mechanism can also be used for distributed systems other than simulation.

Besides local clocks for each component, there is a global clock, indicating a time before which no component has to be rolled back. The progression of the global time for specific models is the subject of several studies.

Kleinrock & Felderman [101] study a discrete-time model with two processors. The local times of the processors increase with geometric jumps and sojourns. After each jump a message for the other processor is generated with a fixed probability. If that processor's local time is ahead of the time of the message, then it rolls back to that time. A related Markov chain is studied and the speed-up, relative to a single processor, is calculated. The results of [101] are a superset of those of Lavenberg et al. [108]. Mitra & Mitrani [124] analyze a model related to that of [101] in which the jump sizes can be arbitrary; their approach is based on a Wiener-Hopf factorization.

In Akyildiz et al. [3] a model with c processors and a limited shared memory capacity is analyzed using a simple Markov process, which approximates the used memory space. The results are compared with experimental data.

4 OPTIMIZATION

In this section, we discuss optimization issues of parallel and distributed systems which can be tackled using queueing network formalisms. We shall first provide a general discussion about load balancing problems. Then we discuss in more detail the routing problem which is a special case of the load balancing problem, followed by a discussion on a dual problem, the problem of server allocation. In the last subsection, we will consider scheduling problems.

4.1 Load balancing

An operational aspect of distributed systems is the availability of a protocol which optimally balances the workload over the servers: a *load balancing protocol.* These protocols can roughly be divided into routing models, where at their moment of arrival in the system jobs are (irrevocably) routed to one of the servers, and server allocation models, where the servers determine from which input sources they draw their jobs.

Another important element of a load balancing protocol is the information it requires to operate. This information can range from total knowledge about the system at any point in time, to only information about some basic characteristics, like arrival rate and service times. In general, the term *dynamic* is used for policies which operate under time dependent information, whereas protocols operating under time independent characteristics of the system are called *static.* Below we give overviews of routing, server allocation and stochastic scheduling models, again with an emphasis on methodology. See Gelenbe & Pekergin [69] for an interesting general discussion on load balancing in parallel and distributed systems, that also touches upon the trade-off between static and dynamic load balancing; see Wang & Morris [164] for a taxonomy of the current load balancing protocols, discriminating between routing (called *source initiative*) and server allocation (*server initiative*) models. They provide numerical comparisons, based on analysis and simulation, of various allocation protocols, both static and dynamic.

4.2 Routing

The routing or customer allocation problem, as a special case of the load balancing problem, consists in assigning arriving customers to one of several parallel queues (which are usually assumed to have a single server). Thus, no jockeying amongst the queues is allowed. We will consider both static and dynamic routing problems.

For all sorts of information structure both the symmetric (i.e., the service times are equally distributed for each queue) and the asymmetric case are studied. For the symmetric models it is often possible to find the optimal policy, mostly using coupling, dynamic programming or stochastic orderings. Consequently, these are transient results, which often hold for a large class of cost functions and general arrivals.

Asymmetric models on the other hand rarely have a simple optimal policy; it usually depends on the arrival process, the service times, etc. The analysis is therefore often numerical in nature, Poisson arrivals are assumed and only long-run results are obtained.

Two static allocation policies have been proposed: probabilistic allocation (assign arriving jobs to a queue according to a fixed probability), and pattern allocation (route arriving jobs to a queue according to a routing table).

When the servers are identical the symmetric (or equal probability) routing policy is optimal among the probabilistic policies for the minimization of response times and resequencing delay. This can be shown using stochastic orderings and coupling (Chang et al. [32], Gün & Jean-Marie [92]).

For the general non-symmetrical problem, Buzen & Chen [30] present an algorithm for determining the probabilistic allocation which minimizes the mean sojourn time of a job.

In most of the numerical studies, queueing theory is used to determine an expression for the performance measure that is to be minimized. The separability of that expression in terms relating to only one particular queue, and the convexity of each term, lead to a tractable non-linear optimization problem of the class of resource allocation problems that is extensively discussed in the book of Ibaraki & Katoh [87].

Ross & Yao [144] study a probabilistic allocation problem with additional dedicated arrival streams and local priority scheduling. Proving convexity in their case is an interesting problem, that is solved using matroid theory. Bonomi & Kumar [22] also discuss probabilistic allocation with additional dedicated arrival streams. They consider the situation where not all system parameters are known, or where some of the parameters may change from time to time. They propose several *adaptive* load balancing algorithms, using stochastic approximation and stochastic control methods.

Pattern allocation leads to a more regular arrival process than probabilistic allocation, and hence better performance can be expected. However, constructing the optimal pattern is generally an unsolved problem. Various studies have been carried out for characterizing the optimal routing policies, see for example [17, 143] and the references therein.

When the servers are identical, Walrand [163] uses coupling arguments to show that assigning the jobs cyclicly to the queues (the round robin policy) is optimal for exponential service times. Recently, using a coupling technique and majorization theory, Liu & Towsley [119] generalized the optimality of the round robin policy to the case of identical IFR (Increasing Failure Rate) servers.

Under the assumption of general service time distributions, the round robin policy yields smaller (in the sense of increasing convex ordering) stationary and transient job waiting times than the symmetric probabilistic routing policy (Stoyan [151], Jean-Marie & Liu [93]).

For the case of non-identical servers Hajek [76] proves how the pattern allocation to a single queue should be, given that a fixed fraction of the arrivals should be sent to that queue, to minimize the average number of customers in that queue. His proof involves showing *multimodularity* (a generalization of convexity to multiple dimensions) of certain functions. Ramakrishnan [138] proposes a useful approximation procedure for non-identical exponential servers; see [44] for the case of general servers. Again separability of the objective function and convexity of each term are exploited.

A well studied dynamic model is that with exponential servers and decisions based on the numbers of jobs in the queues. For the symmetric model the optimality of shortest queue routing was first proved by Winston [167]. This result has been generalized in different directions by various authors. The techniques used are dynamic programming and coupling. A recent paper, showing the optimality for Schur convex cost functions, ILR (i.e., increasing in likelihood ratio) service time distributions and including finite buffers, is [156]. For asymmetric models the optimal policy does not have a nice structure. Several authors have tried to obtain good policies (e.g., Shenker & Weinrib [147]). Using dynamic programming, Hajek [75] showed for the model with two queues (and some additional features) that there is a non-decreasing switching curve. Xu & Chen [171] considered the limiting behavior of this curve for discounted costs, and showed that it converges to a constant, for unequal holding cost rates.

A model with a different information structure is the one where decisions are based on the *workload* in the queues. Routing to the queue with the smallest workload minimizes both the total workload (in fact, each weak Schur convex function of the workload vector is minimized) and the job response times. Note that the smallest workload policy is equivalent to FCFS. Some references are Wolff [168, 169], Foss [61, 62] and Daley [47]. It is interesting to note that basically all results are established using the same coupling argument. A generalization to network models, and an extensive list of references, can be found in [105].

An overview of routing policies and their performances is given by Boel & Van Schuppen [21]. They consider the problem from a control point of view, and discuss the question what amount of information is required at the routing points to achieve good system performance. Their paper concentrates on analytically and numerically tractable models.

4.3 Server allocation

As a dual problem to routing problems (which can be seen as job allocation problems), the problem of server allocation has also received much interest in the literature. However, there are few results on static server allocation problems.

We will restrict ourselves to single server models, one reason being that the results on multiple server models are less relevant to the present survey, the other being that policies, which are optimal for single server models, often perform very well if applied to multiple server models (e.g., Weiss [165]). As a general reference for multiple server models, we refer to Righter [142].

In the simplest model jobs arrive in several queues (single queue models are discussed in Subsection 4.3 on scheduling), each requiring an exponentially distributed amount of processing, depending on the queue. The objective is to minimize the weighted holding costs. Using a simple interchange argument it can be shown that the single server should process the jobs (preemptively) in decreasing order of product of processing rate and holding cost rate [29]. This policy is known as the μc or $c\mu$ rule. Such a type of policy is called a *list policy*, i.e., a policy which has associated a list of the queues, and which processes the job whose queue is highest in the list.

Generalizations are possible in several directions. Assume that the service times are general, and that jobs, after completing service, can re-enter in a, possibly different, queue. We restrict ourselves to Poisson arrivals and non-preemptive policies. The problem of finding the server allocation policy that minimizes the weighted number of jobs in the system is known as *Klimov's problem.* It is shown in [102] that a list policy is optimal. A good reference for Klimov's and related problems is chapter 9 in [163].

If we assume that there are switching times between serving different queues, we arrive at polling models (cf. Subsection 2.5). Polling optimization issues have only recently been tackled. Some static and dynamic server routing optimization problems are reviewed in [24] and [172], respectively. Symmetric optimization models are discussed in [115], and an asymmetric optimization model is studied in [106].

Contrary to most studies discussed so far, the optimal policy in the model of Menich & Serfozo [123] is not a list policy. They augment a symmetric routing model with a movable server, and show that it should be assigned to the longest queue. In the case of finite buffers, the duality between various job allocation and server allocation problems where queue lengths are available to the controller has been established in Sparaggis et al. [149].

4.4 Scheduling

In most routing or server allocation models jobs are processed in FCFS order. Here this and other service policies are discussed. By scheduling we mean policies that determine the order according to which servers serve jobs waiting in the queue.

Consider a single $G/GI/s$ queue. Then the FCFS policy minimizes the stationary waiting times in the sense of the increasing convex ordering, in the case that the service time distribution is of IFR type (Hirayama & Kijima [84], Chang & Yao [35]).

Now assume that every job has a due date. Several papers have studied the effect that different scheduling policies have on the job lateness (defined as the amount of time the completion time of a job exceeds the due date of that job). The optimality of stochastic versions of SDD (the policy which processes jobs with the shortest due date first) has been established in Liu & Towsley [118]. Also for the $G/M/s$ queue when jobs have hard deadlines (meaning that a job leaves the system either when it finishes service or when its due date occurs) SDD is optimal (Towsley & Panwar [155]).

For queueing networks, where each queue has its own servers, the optimality of SDD was first shown in Towsley & Baccelli [154] for queues in tandem in the sense of convex ordering. More general results were established by Liu & Towsley [120] for in-forest networks consisting of multi-server queues. In

[120], extremal properties of FCFS, LCFS (Last Come First Serve), stochastic SDD and LDD (standing for longest due date) policies were proved for the minimization (or maximization) of job response times, lateness and end-to-end delays.

The scheduling problem in more realistic parallel processing models has been addressed in a recent paper by Baccelli, Liu & Towsley [12]. They provide extremal properties of various scheduling strategies for multiprogrammed multitasked multiprocessor systems where task executions are constrained by precedence relations.

In most of the above mentioned studies, the techniques used are stochastic comparison and sample path analysis. Quite strong stochastic qualitative properties are established when these techniques are applicable. A general and unified theoretical formalism of these techniques has been proposed in a recent paper of Liu et al. [117].

In the following studies, the authors use queueing analysis to compare average performance measures of different scheduling policies.

Nelson et al. [34, 121, 122] considered the problem of allocating parallel tasks to processors so as to minimize job (consisting of parallel tasks) response times. They showed that allocation of tasks to different processors is not always a good strategy.

Performance analysis of scheduling policies in multiprogrammed multiprocessor systems with parallel tasks can be found in Nelson et al. [133], [128] for FCFS policies, in Towsley et al. [129] for processor sharing, and in Nelson & Towsley [132] for priority policies. Leutenegger & Vernon [111] provide a comparison of performances of various scheduling policies.

REFERENCES

1. I.J.B.F. Adan. *A Compensation Approach for Queueing Problems.* PhD thesis, Eindhoven University of Technology, 1991.
2. I.J.B.F. Adan, J. Wessels, and W.H.M. Zijm. Analysis of the asymmetric shortest queue problem. *Queueing Systems*, 8:1–58, 1989.
3. I.F. Akyildiz, L. Chen, S.R. Das, R.M. Fujimoto, and R.F. Serfozo. Performance analysis of "time warp" with limited memory. *Performance Evaluation Review*, 20:213–224, 1992.
4. E. Altman, P. Konstantopoulos, and Z. Liu. Stability, monotonicity and invariant quantities in general polling systems. *Queueing Systems*, 11:35–57, 1992.
5. F. Baccelli. Two parallel queues created by arrivals with two demands: The $M/G/2$ symmetrical case. Technical report 426, INRIA-Rocquencourt, 1985.
6. F. Baccelli, G. Balbo, R.J. Boucherie, J. Campos, and G. Chiola. Annotated bibliography on stochastic Petri nets. In O.J. Boxma and G.M. Koole, editors, *Performance Evaluation of Parallel and Distributed Systems — Solution Methods.* CWI, Amsterdam, 1994. CWI Tract.
7. F. Baccelli, E. Gelenbe, and B. Plateau. An end-to-end approach to the resequencing problem. *Journal of the ACM*, 31:474–485, 1984.

8. F. Baccelli, A. Jean-Marie, and Z. Liu. A survey on solution methods for task graph models. In N. Götz, U. Herzog, and M. Rettelbach, editors, *Second QMIPS Workshop*, volume 26(14) of *Arbeitsberichte der IMMD*, Erlangen, 1993.
9. F. Baccelli and T. Konstantopoulos. Estimates of cycle times in stochastic Petri nets. In I. Karatzas and D. Ocone, editors, *Applied Stochastic Analysis*, pages 1–20. Springer Verlag, 1992. Lecture Notes in Control and Information Sciences 177.
10. F. Baccelli and Z. Liu. On the stability condition of a precedence-based queueing discipline. *Journal of Applied Probability*, 21:883–898, 1989.
11. F. Baccelli and Z. Liu. On the execution of parallel programs on multiprocessor systems—a queueing theory approach. *Journal of the ACM*, 37:373–414, 1990.
12. F. Baccelli, Z. Liu, and D. Towsley. Extremal scheduling of parallel processing with and without real-time constraints. *Journal of the ACM*, 40:1209–1237, 1993.
13. F. Baccelli and A.M. Makowski. Synchronization in queueing systems. In H. Takagi, editor, *Stochastic Analysis of Computer and Communication Systems*, pages 57–129. North-Holland, Amsterdam, 1990.
14. F. Baccelli, A.M. Makowski, and A. Shwartz. The fork-join queue and related systems with synchronization constraints: Stochastic ordering and computable bounds. *Advances in Applied Probability*, 21:629–660, 1989.
15. F. Baccelli, W.A. Massey, and D. Towsley. Acyclic fork-join queuing networks. *Journal of the ACM*, 36:615–642, 1989.
16. F. Baccelli and Ph. Robert. Analysis of update response times in a distributed database maintained by the conservative time stamps ordering algorithm. In *Performance '83*, pages 415–436, 1983.
17. C.E. Bell and S. Stidham, Jr. Individual versus social optimization in the allocation of customers to alternative servers. *Management Science*, 29:831–839, 1983.
18. J.P.C. Blanc. A numerical approach to cyclic-service queueing models. *Queueing Systems*, 6:173–188, 1990.
19. J.P.C. Blanc. The power-series algorithm applied to the shortest-queue model. *Operations Research*, 40:157–167, 1992.
20. J.P.C. Blanc. Performance analysis and optimization with the power-series algorithm. In L. Donatiello and R.D. Nelson, editors, *Performance Evaluation of Computer and Communication Systems*, pages 53–80. North-Holland, Amsterdam, 1993.
21. R.K. Boel and J.H. van Schuppen. Distributed routing for load balancing. *Proceedings of the IEEE*, 77:210–221, 1989.
22. F. Bonomi and S. Kumar. Adaptive optimal load balancing in a nonhomogeneous multiserver system with a central job scheduler. *IEEE Transactions on Computers*, 39:1232–1250, 1990.
23. O.J. Boxma. Workloads and waiting times in single-server systems with multiple customer classes. *Queueing Systems*, 5:185–214, 1989.
24. O.J. Boxma. Analysis and optimization of polling systems. In J.W. Cohen

and C.D. Pack, editors, *Queueing, Performance and Control in ATM*, pages 173–183. North-Holland, Amsterdam, 1991.

25. O.J. Boxma and W.P. Groenendijk. Two queues with alternating service and switching times. In O.J. Boxma and R. Syski, editors, *Queueing Theory and its Applications*, pages 261–282. North-Holland, Amsterdam, 1988.
26. O.J. Boxma and G.J. van Houtum. The compensation approach applied to a 2 × 2 switch. *Probability in the Engineering and Informational Sciences*, 7:171–193, 1993. Reprinted in these proceedings.
27. D. Y. Burman and D. R. Smith. A light traffic theorem for multi-server queues. *Mathematics of Operations Research*, 8:15–25, 1983.
28. D. Y. Burman and D. R. Smith. An asymptotic analysis of a queueing system with Markov-modulated arrivals. *Operations Research*, 34:105–119, 1986.
29. C. Buyukkoc, P. Varaiya, and J. Walrand. The $c\mu$ rule revisited. *Advances in Applied Probability*, 17:237–238, 1985.
30. J.P. Buzen and P.P.-S. Chen. Optimal load balancing in memory hierarchies. In J.L. Rosenfeld, editor, *Proceedings of IFIP*, pages 271–275. North-Holland, Amsterdam, 1974.
31. C.-S. Chang. Stability, queue length and delay of deterministic and stochastic queueing networks. *IEEE Transactions on Automatic Control*, to appear 1994.
32. C.-S. Chang, X.L. Chao, and M. Pinedo. A note on queues with Bernoulli routing. In *Proc. 29th Conf. on Decision and Control*, December 1990.
33. C.-S. Chang, P. Heidelberger, S. Juneja, and P. Shahabuddin. The application of effective bandwidth to fast simulation of communication networks. *ACM/IEEE Trans. on Networking*, to appear 1994.
34. C.-S. Chang, R. Nelson, and D.D. Yao. Optimal task scheduling on distributed parallel processors. Technical report, IBM Research Division, Yorktown Heights, 1992.
35. C.-S. Chang and D.D. Yao. Rearrangement, majorization and stochastic scheduling. Technical report RC 16250, IBM Research Division, Yorktown Heights, 1990.
36. J.W. Cohen. On the M/G/2 queueing model. *Stochastic Processes and their Applications*, 12:231–248, 1982.
37. J.W. Cohen. *The Single Server Queue*. North-Holland, Amsterdam, 1982.
38. J.W. Cohen. A two-queue, one-server model with priority for the longer queue. *Queueing Systems*, 2:261–283, 1987.
39. J.W. Cohen. Boundary value problems in queueing theory. *Queueing Systems*, 3:97–128, 1988.
40. J.W. Cohen. *Analysis of Random Walks*. IOS Press, Amsterdam, 1992.
41. J.W. Cohen. On a class of two-dimensional nearest neighbouring random walks. *Journal of Applied Probability*, 1994. To appear.
42. J.W. Cohen. On the analysis of the symmetrical shortest queue. Technical report BS-R9420, CWI, Amsterdam, 1994.
43. J.W. Cohen and O.J. Boxma. *Boundary Value Problems in Queueing*

System Analysis. North-Holland, Amsterdam, 1983.

44. M.B. Combé and O.J. Boxma. Optimization of static traffic allocation policies. *Theoretical Computer Science*, 125:17–43, 1994.
45. J. Dai and Y. Wang. Nonexistence of Brownian models of certain multiclass queueing networks. *Queueing Systems*, 13:41–46, 1993.
46. J. Dai and R. J. Williams. Existence and uniqueness of semimartingale reflecting Brownian motions in convex polyhedrons. Working paper, 1994.
47. D.J. Daley. Certain optimality properties of the first-come first-served discipline for $G|G|s$ queues. *Stochastic Processes and their Applications*, 25:301–308, 1987.
48. A. Dembo and O. Zeitouni. *Large Deviations Techniques and Applications*. Jones and Bartlett Publishers, Boston, 1993.
49. J.-D. Deuschel and D. W. Stroock. *Large Deviations*. Academic Press, San Diego, 1989.
50. R. L. Disney and D. König. Queueing networks: A survey of their random processes. *SIAM Review*, 27:335–403, 1985.
51. P. Downey. Bounds and approximations for overheads in the time to join parallel forks. Working paper, 1994.
52. N. G. Duffield and N. O'Connell. Large deviations and overflow probabilities for the general single-server queue, with applications. Technical report DIAS-STP-93-30, Dublin City University, 1993.
53. P. Dupuis and R. S. Ellis. The large deviation principle for a general class of queueing systems, I. Technical report LCDS 93-10, Brown University, 1993.
54. P. Dupuis and R. S. Ellis. A weak convergence approach to the theory of large deviations. Technical report LCDS 93-6, Brown University, 1993. This manuscript will be published as a book by J. Wiley & Sons.
55. M. Eisenberg. Two queues with alternating service. *SIAM Journal on Applied Mathematics*, 36:287–303, 1979.
56. G. Fayolle. *Methodes Analytiques pour les Files d'Attente Couplees*. PhD thesis, Université de Paris VI, Paris, 1979.
57. G. Fayolle and R. Iasnogorodski. Two coupled processors: The reduction to a Riemann-Hilbert problem. *Z. Wahrsch. Verw. Gebiete*, 47:325–351, 1979.
58. L. Flatto and S. Hahn. Two parallel queues created by arrivals with two demands. *SIAM Journal on Applied Mathematics*, 44:1041–1053, 1984.
59. L. Flatto and H.P. McKean. Two queues in parallel. *Comm. Pure Appl. Math.*, 30:255–263, 1977.
60. G.J. Foschini and J. Salz. A basic dynamic routing problem and diffusion. *IEEE Transactions on Communications*, 26:320–327, 1978.
61. S.G. Foss. Approximation of multichannel queueing systems. *Siberian Mathematical Journal*, 21:851–857, 1981.
62. S.G. Foss. Comparison of servicing strategies in multichannel queueing systems. *Siberian Mathematical Journal*, 22:142–147, 1981.
63. Ch. Fricker and R. Jaibi. Monotonicity and stability of periodic polling models. *Queueing Systems*, 15:211–238, 1994.

64. H.R. Gail, S.L. Hantler, and B.A. Taylor. Spectral analysis of $M/G/1$ type Markov chains. Technical report, IBM Research Division, Yorktown Heights, 1992.
65. E. Gelenbe. *Multiprocessor Performance.* Wiley, New York, 1989.
66. E. Gelenbe. Product-form queueing networks with negative and positive customers. *Journal of Applied Probability*, 28:656–663, 1991.
67. E. Gelenbe. G-networks: A unifying model for neural and queueing networks. *Annals of Operations Research*, 48:433–461, 1994. (Special issue on queueing networks, ed. N.M. van Dijk).
68. E. Gelenbe, P. Glynn, and K. Sigman. Queues with negative arrivals. *Journal of Applied Probability*, 28:245–250, 1991.
69. E. Gelenbe and F. Pekergin. Load balancing pragmatics. Technical report, EHEI Université René Descartes, February 1993.
70. L. Georgiadis and W. Szpankowski. Stability of token passing rings. *Queueing Systems*, 11:7–33, 1992.
71. I. Gertsbakh. The shorter queue problem: A numerical study using the matrix geometric solution. *European Journal on Operations Research*, 15:374–381, 1984.
72. P. W. Glynn. Diffusion approximations. In D.P. Heyman and M.J. Sobel, editors, *Handbook on Operations Research and Management Science, Vol. 2*, pages 145–198. Elsevier Science Publishers B.V., Amsterdam, 1990.
73. W.K. Grassmann. Computational methods in probability theory. In D.P. Heyman and M.J. Sobel, editors, *Handbook on Operations Research and Management Science, Vol. 2*, pages 199–254. Elsevier Science Publishers B.V., Amsterdam, 1990.
74. D. Grillo. Polling mechanism models in communication systems – some application examples. In H. Takagi, editor, *Stochastic Analysis of Computer and Communication Systems*, pages 659–698. North-Holland, Amsterdam, 1990.
75. B. Hajek. Optimal control of two interacting service stations. *IEEE Transactions on Automatic Control*, 29:491–499, 1984.
76. B. Hajek. Extremal splittings of point processes. *Mathematics of Operations Research*, 10:543–556, 1985.
77. S. Halfin. The shortest queue problem. *Journal of Applied Probability*, 22:865–878, 1985.
78. J. M. Harrison and R. J. Williams. Brownian models of open queueing networks with homogeneous customer populations. *Stochastics*, 22:77–115, 1987.
79. J. M. Harrison and R. J. Williams. On the quasireversibility of a multiclass Brownian service station. Working paper, 1989.
80. J. M. Harrison, R. J. Williams, and H. Chen. Brownian models of closed queueing networks with homogeneous customer populations. *Stochastics*, 29:37–74, 1990.
81. P.G. Harrison and E. Pitel. Response time distributions in tandem G-networks. In O.J. Boxma and G.M. Koole, editors, *Performance Evaluation of Parallel and Distributed Systems — Solution Methods.* CWI,

Amsterdam, 1994. CWI Tract.

82. G. Harrus and B. Plateau. Queueing analysis of a reordering issue. *IEEE Transactions on Software Engineering*, 8:113–122, 1982.
83. P. Heidelberger and K.S. Trivedi. Queueing network models for parallel processing with asynchronous tasks. *IEEE Transactions on Computers*, 31:1099–1109, 1982.
84. T. Hirayama and M. Kijima. An extremal property of FIFO discipline in $G/IFR/1$ queues. *Advances in Applied Probability*, 21:481–484, 1989.
85. G. Hooghiemstra, M. Keane, and S. van de Ree. Power series for stationary distributions of coupled processor models. *SIAM Journal on Applied Mathematics*, 48:1159–1166, 1988.
86. R. Iasnogorodski. *Problèmes-Frontières dans les Files d'Attente.* PhD thesis, Université de Paris VI, Paris, 1979.
87. T.I. Ibaraki and N. Katoh. *Resource Allocation Problems.* MIT Press, Cambridge, 1988.
88. I. Iliadis and L.Y.-C. Lien. Resequencing delay for a queueing system with two heterogeneous servers under a threshold-type scheduling. *IEEE Transactions on Communications*, 36:692–702, 1988.
89. A. Jean-Marie. *Modélisation de réseaux d'interconnexion multi etages.* PhD thesis, Université Paris Sud, 1987.
90. A. Jean-Marie. Load balancing in a system of two queues with resequencing. In P.J. Courtois and G. Latouche, editors, *Performance '87*, pages 75–88. North-Holland, Amsterdam, 1988.
91. A. Jean-Marie. Analytic computation of Lyapunov exponents in stochastic event graphs. In O.J. Boxma and G.M. Koole, editors, *Performance Evaluation of Parallel and Distributed Systems — Solution Methods.* CWI, Amsterdam, 1994. CWI Tract.
92. A. Jean-Marie and L. Gün. Parallel queues with resequencing. *Journal of the ACM*, 40:1188–1208, 1993.
93. A. Jean-Marie and Z. Liu. Stochastic comparisons for queueing models via random sums and intervals. *Advances in Applied Probability*, 24:960–985, 1992.
94. D.R. Jefferson. Virtual time. *ACM Transactions on Programming Languages and Systems*, 7:404–425, 1985.
95. F. Kamoun, L. Kleinrock, and R. Muntz. Queueing analysis of a reordering issue in the distributed database concurrency control mechanism. In *Proceedings of the 2nd International Conference on Distributed Computing Systems*, pages 13–23, 1982.
96. F.P. Kelly. *Reversibility and Stochastic Networks.* Wiley, New York, 1979.
97. C. Kim and A.K. Agrawala. Analysis of the fork-join queue. *IEEE Transactions on Computers*, 38:250–255, 1989.
98. J.F.C. Kingman. Two similar queues in parallel. *Annals of Mathematical Statistics*, 32:1314–1323, 1961.
99. S.J. de Klein. *Fredholm Integral Equations in Queueing Analysis.* PhD thesis, University of Utrecht, 1988.
100. L. Kleinrock. Performance evaluation of distributed computer-

communication systems. In O.J. Boxma and R. Syski, editors, *Queueing Theory and its Applications*, pages 1–57. North-Holland, Amsterdam, 1988.

101. L. Kleinrock and R.E. Felderman. Two processor time warp analysis: A unifying approach. *International Journal in Computer Simulation*, 2:345–371, 1992.
102. G.P. Klimov. Time-sharing service systems I. *Theory of Probability and its Applications*, 19:532–551, 1974.
103. C. Knessl, B.J. Matkowsky, Z. Schuss, and C. Tier. Two parallel queues with dynamic routing. *IEEE Transactions on Communications*, 34:1170–1176, 1986.
104. C. Knessl, B.J. Matkowsky, Z. Schuss, and C. Tier. Two parallel $M/G/1$ queues where arrivals join the system with the smaller buffer content. *IEEE Transactions on Communications*, 35:1153–1158, 1987.
105. G.M. Koole. On the optimality of FCFS for networks of multi-server queues. Technical report BS-R9235, CWI, Amsterdam, 1992.
106. G.M. Koole. Assigning a single server to inhomogeneous queues with switching costs. Technical report BS-R9405, CWI, Amsterdam, 1994.
107. G.M. Koole. On the power series algorithm. In O.J. Boxma and G.M. Koole, editors, *Performance Evaluation of Parallel and Distributed Systems — Solution Methods.* CWI, Amsterdam, 1994. CWI Tract.
108. S. Lavenberg, R. Muntz, and B. Samadi. Performance analysis of a rollback method for distributed simulation. In *Performance '83*, pages 117–132, 1983.
109. S.S. Lavenberg. A perspective on queueing models of computer performance. In O.J. Boxma and R. Syski, editors, *Queueing Theory and its Applications*, pages 59–94. North-Holland, Amsterdam, 1988.
110. K.K. Leung. Waiting time distributions for token-passing systems with limited-k service via discrete Fourier transforms. In P.J.B. King, I. Mitrani, and R.J. Pooley, editors, *Performance '90*, pages 333–347. North-Holland, Amsterdam, 1990.
111. S.T. Leutenegger and M.K. Vernon. The performance of multiprogrammed multiprocessor scheduling policies. Technical Report TR 913, University of Wisconsin-Madison, 1990.
112. H. Levy and M. Sidi. Polling systems: Applications, modeling and optimization. *IEEE Transactions on Communications*, 38:1750–1760, 1990.
113. Z. Liu and F. Baccelli. Generalized precedence-based queueing systems. *Mathematics of Operations Research*, 17:615–639, 1992.
114. Z. Liu and P. Nain. Sensitivity results in open, closed, and mixed product-form queueing networks. *Performance Evaluation*, 13:237–251, 1991.
115. Z. Liu, P. Nain, and D. Towsley. On optimal polling policies. *Queueing Systems*, 11:59–83, 1992.
116. Z. Liu, P. Nain, and D. Towsley. Exponential bounds with an application to call admission. Technical report, University of Massachusetts, 1994.
117. Z. Liu, P. Nain, and D. Towsley. Sample path methods in the control of queues. *Queueing Systems*, 1994. To appear.

118. Z. Liu and D. Towsley. Effects of service disciplines in $G/G/s$ queueing systems. *Annals of Operations Research*, 48:401–429, 1994. (Special issue on queueing networks, ed. N.M. van Dijk).

119. Z. Liu and D. Towsley. Optimality of the round robin routing policy. *Journal of Applied Probability*, 1994. To appear.

120. Z. Liu and D. Towsley. Stochastic scheduling in in-forest networks. *Advances in Applied Probability*, 26:222–241, 1994.

121. A.M. Makowski and R. Nelson. Distributed parallelism considered harmful. Technical report, IBM Research Division, Yorktown Heights, 1991.

122. A.M. Makowski and R. Nelson. Optimal scheduling for a distributed parallel processing model. Technical report, IBM Research Division, Yorktown Heights, 1992.

123. R. Menich and R.F. Serfozo. Optimality of routing and servicing in dependent parallel processing systems. *Queueing Systems*, 9:403–418, 1991.

124. D. Mitra and I. Mitrani. Analysis and optimum performance of two message-passing parallel processors synchronized by rollback. In E. Gelenbe, editor, *Performance '84*, pages 35–50, 1984.

125. I. Mitrani. *Simulation Techniques for Discrete Event Systems.* Cambridge University Press, Cambridge, 1982.

126. I. Mitrani and D. Mitra. A spectral expansion method for random walks on semi-infinite strips. In R. Beauwens and P. de Groen, editors, *Iterative Methods in Linear Algebra.* North-Holland, Amsterdam, 1992.

127. T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77:541–580, 1989.

128. R. Nelson. A performance evaluation of a general parallel processing model. *Performance Evaluation Review*, 18:13–26, 1990.

129. R. Nelson, G. Rommel, and J.A. Stankovic. Analysis of fork-join program response times on multiprocessors. *IEEE Transactions on Parallel and Distributed Systems*, 1:286–303, 1990.

130. R. Nelson and A.N. Tantawi. Approximate analysis of fork/join synchronization in parallel queues. *IEEE Transactions on Computers*, 37:739–743, 1988.

131. R. Nelson and A.N. Tantawi. Approximating task response times in fork/join queues. In E. Gelenbe, editor, *High Performance Computer Systems*, pages 157–167. Elsevier Science Publishers B.V., Amsterdam, 1988.

132. R. Nelson and D. Towsley. A performance evaluation of several priority policies for parallel processing systems. *Journal of the ACM*, 40:714–740, 1993.

133. R. Nelson, D. Towsley, and A.N. Tantawi. Performance analysis of parallel processing systems. *IEEE Transactions on Software Engineering*, 14:532–540, 1988.

134. R.D. Nelson and T.K. Philips. An approximation for the mean response time for shortest queue routing with general interarrival and service times. *Performance Evaluation*, 17:123–139, 1993.

135. M.F. Neuts. *Matrix-Geometric Solutions in Stochastic Models.* Johns Hopkins, Baltimore, 1981.

136. V. Nguyen. Processing networks with parallel and sequential tasks: Heavy traffic analysis and Brownian limits. *Annals of Applied Probability*, 3:28–55, 1993.
137. V. Nguyen. The trouble with diversity: Fork-join networks with heterogeneous customer population. *Annals of Applied Probability*, 4:1–25, 1994.
138. K.K. Ramakrishnan. *The Design and Analysis of Resource Allocation Policies in Distributed Systems.* PhD thesis, University of Maryland, Dept. of Computer Science, 1983.
139. M. I. Reiman and B. Simon. Open queueing networks in light traffic. *Mathematics of Operations Research*, 14:26–59, 1989.
140. M. I. Reiman, R. J. Williams, and H. Chen. A boundary property of semimartingale reflecting Brownian motions. *Probability Theory and Related Fields*, 77:87–97, 1988.
141. J.A.C. Resing. Polling systems and multitype branching processes. *Queueing Systems*, 13:409–426, 1993.
142. R. Righter. Scheduling. In M. Shaked and J.G. Shanthikumar, editors, *Stochastic Orders and their Applications.* Academic Press, New York, 1994.
143. Z. Rosberg. Deterministic routing to buffered channels. *IEEE Transactions on Communications*, 34:504–507, 1986.
144. K.W. Ross and D.D. Yao. Optimal load balancing and scheduling in a distributed computer system. *Journal of the ACM*, 38:676–690, 1991.
145. R.Y. Rubinstein. *Monte Carlo Optimization, Simulation and Sensitivity of Queueing Networks.* Wiley, New York, 1986.
146. C.H. Sauer and E.A. MacNair. *Simulation of Computer Communication Systems.* Prentice-Hall, Englewood Cliffs, 1983.
147. S. Shenker and A. Weinrib. The optimal control of heterogeneous queueing systems: A paradigm for load-sharing and routing. *IEEE Transactions on Computers*, 38:1724–1735, 1989.
148. A. Shwartz and A. Weiss. Induced rare events: Analysis via large deviations and time reversal. *Advances in Applied Probability*, 25:667–689, 1993.
149. P.D. Sparaggis, C.G. Cassandras, and D. Towsley. On the duality between routing and scheduling systems with finite buffer space. *IEEE Transactions on Automatic Control*, 38:1440–1446, 1993.
150. W.J. Stewart, editor. *Numerical Solution of Markov Chains.* Marcel Dekker, New York, 1991.
151. D. Stoyan. *Comparison Methods for Queues and Other Stochastic Models.* Wiley, New York, 1984.
152. H. Takagi. Queueing analysis of polling models: An update. In H. Takagi, editor, *Stochastic Analysis of Computer and Communication Systems*, pages 267–318. North-Holland, Amsterdam, 1990.
153. H. Takagi. Applications of polling models to computer networks. *Computer Networks and ISDN Systems*, 22:193–211, 1991.
154. D. Towsley and F. Baccelli. Comparison of service disciplines in a tandem queueing network with delay dependent customer behavior. *Operations*

Research Letters, 10:49–55, 1991.

155. D. Towsley and S.S. Panwar. Optimality of the stochastic earliest deadline policy for the $G/M/c$ queue serving customers with deadlines. COINS Technical Report 91-61, University of Massachusetts at Amherst, 1991.
156. D. Towsley and P.D. Sparaggis. Optimal routing in systems with ILR service time distributions. CMPSCI Technical Report 93-13, University of Massachusetts at Amherst, 1993.
157. P. Tsoucas. Rare events in series of queues. *Journal of Applied Probability*, 1994. To appear.
158. N.M. van Dijk. *Queueing Networks and Product Forms.* Wiley, New York, 1993.
159. G.J.J.A.N. van Houtum, I.J.B.F. Adan, J. Wessels, and W.H.M. Zijm. The compensation approach for three and more dimensional random walks. Technical Report COSOR 92-39, Eindhoven University of Technology, Eindhoven, 1992.
160. S. Varma. *Heavy and Light Traffic Approximations for Queues with Synchronization Constraints.* PhD thesis, University of Maryland, 1990.
161. S. Varma. Performance evaluation of the time-stamp ordering algorithm in a distributed data base. *IEEE Transactions on Parallel and Distributed Systems*, 4:668–676, 1993.
162. S. Varma and A.M. Makowski. Interpolation approximations for symmetric fork-join queues. In *Performance '93*, pages 245–273, 1993.
163. J. Walrand. *An Introduction to Queueing Networks.* Prentice-Hall, Englewood Cliffs, 1988.
164. Y.T. Wang and R.J.T. Morris. Load sharing in distributed systems. *IEEE Transactions on Computers*, 34:204–217, 1985.
165. G. Weiss. Approximation results in parallel machines stochastic scheduling. *Annals of Operations Research*, 26:195–242, 1990.
166. W. Whitt. The amount of overtaking in a network of queues. *Networks*, 14:411–426, 1984.
167. W. Winston. Optimality of the shortest line discipline. *Journal of Applied Probability*, 14:181–189, 1977.
168. R.W. Wolff. An upper bound for multi-channel queues. *Journal of Applied Probability*, 14:884–888, 1977.
169. R.W. Wolff. Upper bounds on work in system for multichannel queues. *Journal of Applied Probability*, 24:547–551, 1987.
170. P.E. Wright. Two parallel processors with coupled inputs. *Advances in Applied Probability*, 24:986–1007, 1992.
171. S.H. Xu and H. Chen. A note on the optimal control of two interacting service stations. Working paper, 1991.
172. U. Yechiali. Optimal dynamic control of polling systems. In J.W. Cohen and C.D. Pack, editors, *Queueing, Performance and Control in ATM*, pages 205–217. North-Holland, Amsterdam, 1991.
173. Y. Zhao and W.K. Grassmann. The shorter queue problem: a numerical study using the matrix geometric solution. *Queueing Systems*, 8:59–79, 1991.

Annotated Bibliography on Stochastic Petri Nets*

F. Baccelli[†], G. Balbo[‡], R. Boucherie[†], J. Campos[§], G. Chiola[‡]
[†] *INRIA Sophia-Antipolis, 06902 Sophia-Antipolis, France*
[‡] *Università di Torino, 10149 Torino, Italy*
[§] *Universidad de Zaragoza, 50015 Zaragoza, Spain*

An annotated bibliography on stochastic Petri nets is given.

1 DEFINITION

Originally, no notion of time was included in the definition of Petri nets and their use was limited to the study of the logical (qualitative) properties of systems. Their application to the analysis of real systems soon made clear the necessity of incorporating time specifications in the definition of the formalism to make it useful for the assessment of the performance (quantitative evaluation) of systems. Several authors proposed the association of time with the nodes of the models described with the Petri net formalism. Among these first proposals are worth mentioning those due to Noe and Nutt [172], Ramchandany [175], Merlin and Farber [165].

The idea of associating a random delay with the firing of the transitions was independently proposed by several authors [48, 169, 170, 184] and led to the definition of Stochastic Petri Nets (SPN) [117, 168] in which a Markov chain that is isomorphic to the state space of the net can be defined starting from the time specifications of the model. Alternative definitions of timed Petri nets associated with underlying stochastic processes are also due to [190, 176, 192].

The use of Stochastic Petri Nets for the performance analysis of interesting problems coming mostly from the area of computer architecture, led to the definition of Generalized Stochastic Petri Nets (GSPN) in which the transitions may be of two different types (timed and immediate) depending on whether a delay is specified for their firing or not [14]. Subsequently, Extended Stochastic Petri Nets were proposed in which the most important additional feature is represented by the presence of probabilistic arcs that upon firing of a transition may deposit tokens on subsets of its output set depending on a probability distribution [111]. Similar ideas are also contained in the Stochastic Activity Nets [166] that have been developed for the evaluation of systems affected by failures.

Trying to understand the deep implications that the extensions proposed by these last models have on the qualitative as well as quantitative properties of this generalized formalism, several papers have been published in which different aspects of the problem have been discussed [5, 3, 10, 7, 49, 73, 98]. The

---

*Supported by the European Grant BRA-QMIPS of CEC DG XIII.

current definition of the GSPN formalism is contained in [86] and characterizes a modelling tool that is now well understood and well established.

Dealing with increasingly complex models has led several researchers to propose additional extensions of the basic stochastic Petri net formalism including the idea of colored tokens (Colored Stochastic Petri Nets - CSPN) [191] and of controlling transition firings by means of boolean expressions introduced in the models at the moment of their specification (Predicate/Transition Stochastic Petri Nets - P/TSPN) [159, 160]. In order to keep under control the growth of the complexity of the stochastic process underlying these models and to obtain a modelling formalism that is well suited for the representation of highly symmetrical systems, some restrictions on the specification of the colors of the tokens and of the functions that are used for their manipulation have been introduced in the definition of CSPN, leading to the proposal of Well Formed Stochastic Petri Nets that are today one of the most used modelling tools for the high-level description of complex systems [113, 81, 82, 85].

2 NUMERICAL SOLUTION

In the early proposals of the SPN modelling technique, the Markovian numerical analysis based on the construction of the Reachability Graph was adopted as the standard analysis tool [12, 90, 110, 102, 179, 122]. The technique is relatively easy to implement and to use if properly embedded into a tool with a good graphic interface that allows the user to ignore the mathematical details of the technique and concentrate on the examination of the performance results represented at the net level [167, 89].

Unfortunately the application of the technique is severely limited by the memory requirements of the Reachability Graph computation algorithm and the complexity of the numerical solution of the Markov chain. Early attempts to address the size problem include the idea of using approximate hybrid solutions exploiting computationally efficient algorithms based on product form queueing networks [41, 42, 43]. The technique has been subsequently applied also in [56] and related to hierarchical model decomposition in the framework of colored net models [55].

Another point of interest in the case of GSPN models comprising immediate transitions is the elimination of the *vanishing* states (the ones in which the model spends zero time due to the enabling of immediate transitions). Earlier algorithms performed this elimination globally [14, 97, 96]. A time decomposition approach was proposed in [24]. On the fly elimination based on static identification of extended conflict sets where proposed in [39] and implemented in [94] yielding substantial advantage in terms of memory space for models with large numbers of vanishing states. In general, the idea of static structural analysis of GSPN models to compute properties [86] that can be exploited at run time to reduce the complexity of the Reachability Graph enumeration algorithm [92] is the key for the success of the GreatSPN tool in its present form [94]. For the transient analysis, the *randomization* algorithm has proven to be superior to other algorithms previously used [161].

In the case of colored Petri net models, intrinsic symmetries can be exploited to reduce the cost and complexity of the numerical solution algorithms [191, 160, 83, 68]. In particular, lumping techniques can be automatically exploited in the case of *well-formed* net models [112, 81, 82]. Prototype implementations of the technique [87] have already proven the tremendous gain that can be obtained in case of highly symmetric models [85].

Another way of extending the feasibility of numerical analysis for SPNs and GSPNs yielding a huge state space is to resort to parallel processing. In [70] a first attempt to use a massively parallel SIMD architecture for the generation of the Reachability Graph of GSPNs is described. One order of magnitude gain in the number of states that can be handled is claimed for a Connection Machine CM-2 with respect to powerful workstations, so that models with several millions of states can be analyzed. Kronecker algebra is used in [108, 107] to guide the implementation of a distributed version of the Markovian solution on Transputer based MIMD architectures.

The numerical analysis of open (i.e., unbounded) SPN models has been addressed in [120], where a technique is proposed in the case of SPNs with a single unbounded place. An original matrix method for closed SPN models has been proposed in [118] as an alternative to the usual Markovian numerical approach. Its viability in terms of CPU time and memory requirements with respect to the usual approach has not been demonstrated yet, however.

3 PRODUCT FORM RESULTS

Product form results for the equilibrium distribution of stochastic Petri nets have been derived first for some special cases [6, 91, 118] and subsequently for a few classes of stochastic Petri nets by analogy with results for queueing networks (Baskett *et al.* [46], Gordon and Newell [128], Jackson [141], Kelly [147], Serfozo [182], Van Dijk [186], Whittle [189]). These results are closely related but not equivalent due to the incorporation of structural properties (e.g., T-invariants) in the product form results for stochastic Petri nets. Similar to the results for queueing networks, a product form stochastic Petri net offers enormous computational advantages.

Product form results for stochastic Petri nets can be separated into three classes. The first set of results covers Petri nets in which in each transition a single token can move from one place to another. The second set of results allows multiple tokens to move in each transition. The equilibrium distribution in these results can usually be presented as a product over the places of the Petri net. The third set of results covers Petri nets for which the equilibrium distribution is a product over subnetworks.

Li and Woodside [155] present product form results for *state machines*, the equivalent for queueing networks in the Petri net formalism, and for *serial-parallel* Petri nets, obtained by adding places to a state machine such that the underlying Markov chain remains isomorphic to the Markov chain for the state machine. Lazar and Robertazzi [153], [152], [178] obtain product forms for *safe* stochastic Petri nets that are comprised of *task sequences sharing common buffers.* These results require the state space of the underlying Markov chain to be a multidimensional toroidal manifold. These results are extended and formalized by Frosch [126], [124], [125]. The framework of Frosch is that of *synchronized systems of sequential processes*: state machines that share buffer places. The evolution of these nets resembles the evolution of state machines.

The Petri net equivalent of batch routing queueing networks is analyzed by Henderson *et al.* [134], [137]. In these nets multiple tokens are involved in each transition. Boucherie and Sereno [51] show that these Petri nets can be characterized via *minimal closed support T-invariants.* Coleman *et al.* [101] provide sufficient conditions for the equilibrium distribution to be a product of Jackson-type (a product over places).

The Petri nets of Henderson *et al.* and Frosch are mainly disjoint classes of nets that share state machines only. A comparison between these classes is given by Donatelli and Sereno [106]. Boucherie and Sereno [52] extend and unify these two classes via the framework of batch routing queueing networks with state-dependent routing (Boucherie and van Dijk [53]). This generalization incorporates inhibitor arcs in the product form formalism, and allows general marking dependent firing rates, and marking dependent enabling of transitions (as long as the product form conditions are satisfied).

Petri nets for which the equilibrium distribution can be expressed as a *product over subnetworks* are analyzed by Boucherie [50], Henderson and Lucic [132], and Li and Georganas [158]. Boucherie [50] characterizes the product form via independence arguments for *competing Markov chains* representing Petri nets that are synchronized via buffer places. Henderson and Lucic [132] and Li and Georganas [158] use their product over subnets to obtain computational schemes related to aggregation of subnets.

The normalization constant for product form Petri nets is computed in Coleman [100], via a recursion in the number of places, and in Coleman *et al.* [101], Sereno and Balbo [180] via a recursion in the number of tokens (convolution algorithm) as well as in the number of places (Mean Value Analysis algorithm) [181]. The theoretical basis of this last approach is proposed in [36].

4 BOUNDS

There are numerous techniques for deriving bounds in stochastic Petri nets.

A first approach consists in the computation of insensitive (i.e. valid for all distribution functions) upper and lower bounds for the performance indices of timed Petri nets, based on linear programming techniques, net p-invariants, and Little's law. This approach was followed by Campos, Chiola and Silva in [63], [64], [62], [66] and [59]. This technique can be improved when something is known on the distribution of the activities as shown in [57] and [60]. It can also be extended to the case of well-formed colored Petri nets [80].

A second one is based on the derivation of upper and lower bounds for the conditional token probabilities in a subnet of a stochastic Petri net, from probabilistic arguments. This technique can be used to bound the error due to aggregation and time scale decomposition of nets (it is applicable to systems containing activities the durations of which differ by several orders of magnitude). This approach was followed by Campos, Silva and coauthors in [140].

A third technique consists in using stochastic graph representations of the net of interest in order to derive bounds. This was done by Rajsbaum in [174] in order to derive upper and lower bounds for the throughput of stochastic marked graphs with exponential, independent, and identically distributed random firing times. For marked graphs, other approaches based on stochastic recursions were developed by Baccelli and Liu in [31] using convex ordering and by Baccelli and Konstantopoulos using large deviations [30].

5 STABILITY

A first line of research concerns the computation of saturation conditions for particular classes of stochastic Petri nets with exponential and independent random times. The first papers on the matter were by Florin and Natkin [121], [119]. See also the paper by Campos and Silva [58]. Necessary and sufficient saturation conditions for stochastic Petri nets with general distributions, were also derived in some particular cases using techniques from stochastic processes (in particular, martingales theory) by Campos, Plo and San Miguel [61], [129], and by Baccelli, Bambos, and Walrand [34].

The approach by recursive equations which was developed for stochastic marked graphs by Baccelli [35] and for nets with switching by Baccelli, Cohen and Gaujal [33], allows for a complete analysis of stability (see the book by Baccelli, Cohen, Olsder and Quadrat [32] for marked graphs and the paper by Baccelli and Gaujal for free choice nets [29]). The technique also allows one to determine coupling times as shown in [151].

6 ANALYTICAL RESULTS

Outside product form results, the main analytical results are based on the $(\max, +)$ approach. This approach was first used for deterministic marked graphs by Cohen, Dubois, Quadrat and Viot in [99], where the periodic regimes of event graphs are understood as spectral properties of matrices in this algebra. The extension to stochastic marked graphs was considered in [35]. A classification of the stationary regimes and necessary and sufficient conditions for their uniqueness was proposed by Mairesse in [163, 162]. A Markov chain analysis based on this representation was considered by Olsder, Resing, de Vries, Keane and Hooghiemstra in [173] and by Jean-Marie [142]. A survey and a bibliography on this approach can be found in the book by Baccelli, Cohen, Olsder and Quadrat [32].

7 APPROXIMATIONS

There is a large number of approximation techniques for stochastic Petri nets:

- Flow equivalent approximation techniques [144];
- Iterative approximation techniques for subclasses of stochastic Petri nets, based on decomposition and aggregation [156, 148, 149, 146, 65, 104, 145, 157];

- Iterative approximation techniques for subclasses of stochastic Petri nets, based on independence or near-independence of the underlying Markov chain (orthogonal concepts to decomposability and near-decomposability) [95];
- Approximative aggregation techniques for hierarchical well-formed colored nets, based on the regular structure of the generator matrix of the underlying MC [55, 54, 123].

8 DISCRETE EVENT SIMULATION

Simulation was one of the earliest techniques proposed for the analysis of timed Petri net models [172]. It was introduced as a viable, practical way of dealing with time extensions that do not allow for exact numerical analysis [111, 130, 131].

In another approach, simulation was seen as the alternative to Markovian analysis in case of models with huge state space that prevents numerical solution on computer systems due to memory shortage [90, 89]. In this case efficiency is a crucial issue in order to handle large models not amenable to other analysis techniques. The idea was then pursued to try and exploit preliminary structural analysis to improve the efficiency of the simulation engine for SPN models [92, 37, 94, 93].

The next step has been the exploitation of model symmetries naturally described by the Well-formed Colored net formalism to speed-up simulation as well as numerical solution [87]. This approach led to the introduction of the concept of *symbolic* simulation for colored nets [84, 79]. According to this technique the average length of the list of scheduled events is kept smaller by scheduling only the first event per equivalence class up to a symmetric permutation of basic colors.

In case of large SPN models an intuitive way of speeding up the simulation is to apply classical *distributed simulation* techniques to a static partition of the model [127]. Both the conservative [72] and the optimistic [143] methods can be adapted to the simulation of SPNs [185, 171, 25, 114]. Also in this case the exploitation of statically precomputed structural properties seems to be the key for improving the performance and obtaining real speedup compared to sequential simulation [76, 74, 75]. In some particular cases such as for example Marked Graphs the gain can be dramatic [115]. A new distributed simulation protocol with intermediate characteristics with respect to the conservative and the optimistic ones that dynamically exploits lookahead information computed based on structural net properties has been proposed in the SPN framework [78]. Its implementation on the Connection Machine CM-5 is currently in progress [77].

Different simulation methods based on the recursive equation approach were recently derived by Baccelli and Canales in [28]. This method allows for a SIMD simulation of Event Graphs, and yields significant speed-up (up to three orders of magnitude when executed on a CM 2). More on this technique can be found in the thesis of M. Canales [67].

9 APPLICATIONS

The use of stochastic Petri nets for the evaluation of interesting systems is reported in numerous papers that can be found in the proceedings of many international conferences and in important international journals. The application fields in which SPNs have proven to be successful are listed in [9, 8] where some simple examples of models analyzed with this technique can also be found. An exhaustive list of references of this type is difficult to produce, but some papers are mentioned in the rest of this section to point out the type of applications for which the use of SPNs turned out to be important.

The interconnection structure of both loosely and tightly coupled multiprocessor systems has been widely studied with the help of SPNs [15, 13, 16, 2, 17, 11, 20, 69, 71, 1, 27, 85]. SPNs have also been employed for the evaluation of strategies in the use of these complex systems [109, 183], for the analysis of configurations comprising clusters of computers [139], and for the analysis of specific architectural features such as the Floating Point Unit of a particular computer [47].

The problem of writing reliable and efficient concurrent software has been studied with the help of SPNs showing that qualitative and quantitative analysis of specific algorithms can be performed by means of the same model [40] and that the analysis of concurrent software can be done with this technique to identify first the exact structure of the application and to subsequently allow the efficient allocation of processes on the computational nodes of a parallel architecture [103, 45, 44, 116].

Another important field of application is that of communication systems where several communication protocols have been analyzed by means of SPN models [105, 18, 19, 138, 21, 22, 88, 150].

Flexible Manufacturing Systems have also been studied with the help of SPNs showing that a formalism and a technique originally developed for the analysis of parallel and distributed computing systems can be conveniently employed also in application fields that are apparently quite different, but that present instead quite a lot of commonality [23, 38, 154, 187, 164].

Finally SPNs have been found useful in the analysis of systems affected by failures as it is shown in [4, 26].

REFERENCES

1. D. Abbati, S. Caselli, G. Conte, and F. Zanichelli. Synthesis of GSPN models for workload mapping on concurrent architectures. In *Proc. 5th Intern. Workshop on Petri Nets and Performance Models*, Toulouse, France, October 1993. IEEE-CS Press.
2. M. Ajmone Marsan and G. Chiola. Construction of generalized stochastic Petri net models of bus oriented multiprocessor systems by stepwise refinements. In *Proc. 2^{nd} Int. Conf. on Modeling Techniques and Tools for Performance Analysis*, Sophia Antipolis, France, June 1985. ACM.
3. M. Ajmone Marsan and G. Chiola. On Petri nets with deterministic and exponentially distributed firing times. In *Proc. 7^{th} European Workshop on*

Application and Theory of Petri Nets, pages 151–165, Oxford, England, June 1986. Reprinted in G. Rozenberg, ed., Advances on Petri Nets '87, LNCS 266, pp.132–145, Springer Verlag, 1987.

4. M. Ajmone Marsan, A. Bobbio, G. Conte, and A. Cumani. Performance analysis of degradable multiprocessor systems using generalized stochastic Petri nets. *IEEE-CS Distributed Processing T-C Newsletters*, 1(6), 1984.
5. M. Ajmone Marsan, G. Balbo, A. Bobbio, G. Chiola, G. Conte, and A. Cumani. On Petri nets with stochastic timing. In *Proc. International Workshop on Timed Petri Nets*, pages 80–87, Torino, Italy, July 1985. IEEE-CS Press.
6. M. Ajmone Marsan, G. Balbo, G. Chiola, and S. Donatelli. On the product form solution of a class of multiple bus multiprocessor system models. *Journal of Systems and Software*, 6(1,2):117–124, May 1986.
7. M. Ajmone Marsan, G. Balbo, A. Bobbio, G. Chiola, G. Conte, and A. Cumani. The effect of execution policies on the semantics and analysis of stochastic Petri nets. *IEEE Transactions on Software Engineering*, 15(7):832–846, July 1989.
8. M. Ajmone Marsan, G. Balbo, G. Chiola, G. Conte, S. Donatelli, and G. Franceschinis. An introduction to Generalized Stochastic Petri Nets. *Microelectronics and Reliability*, 31(4):699–725, 1991. Special issue on Petri nets and related graph models.
9. M. Ajmone Marsan, G. Balbo, G. Chiola, and G. Conte. Applicability of stochastic Petri nets to performance modeling. In G. Iazeolla, P.J. Courtois, and O.J. Boxma, editors, *Proc. 2^{nd} Int. Workshop Applied Mathematics & Performance/Reliability Models of Computer/Communication Systems*, Roma, Italy, May 1987. North-Holland.
10. M. Ajmone Marsan, G. Balbo, G. Chiola, and G. Conte. Generalized stochastic Petri nets revisited: Random switches and priorities. In *Proc. Int. Workshop on Petri Nets and Performance Models*, pages 44–53, Madison, WI, USA, August 1987. IEEE-CS Press.
11. M. Ajmone Marsan, G. Balbo, G. Chiola, and G. Conte. Modeling the software architecture of a prototype parallel machine. In *Proc. 1987 SIGMETRICS Conference*, Banf, Alberta, Canada, May 1987. ACM.
12. M. Ajmone Marsan, G. Balbo, G. Ciardo, and G. Conte. A software tool for the automatic analysis of generalized stochastic Petri net models. In *Proc. International Conference on Modelling Techniques and Tools for Performance Analysis*, Paris, France, May 1984.
13. M. Ajmone Marsan, G. Balbo, G. Conte, and F. Gregoretti. Modeling bus contention and memory interference in a multiprocessor system. *IEEE Transactions on Computers*, 32(1):60–72, January 1983.
14. M. Ajmone Marsan, G. Balbo, and G. Conte. A class of generalized stochastic Petri nets for the performance analysis of multiprocessor systems. *ACM Transactions on Computer Systems*, 2(1), May 1984.
15. M. Ajmone Marsan, G. Balbo, and G. Conte. Comparative performance analysis of single bus multiprocessor architectures. *IEEE Transactions on Computers*, 31(12), December 1982.

16. M. Ajmone Marsan, G. Balbo, and G. Conte. Modeling single bus multiprocessor systems with external shared resources. In *Proc.* 15^{th} *Annual Pittsburg Conference on Modeling and Simulation*, Pittsburg, USA, April 1984.
17. M. Ajmone Marsan, G. Balbo, and G. Conte. *Performance Models of Multiprocessor Systems.* MIT Press, Cambridge, USA, 1986.
18. M. Ajmone Marsan, G. Chiola, and A. Fumagalli. An accurate performance model of CSMA/CD bus LAN. In *Proc.* 7^{th} *European Workshop on Application and Theory of Petri Nets*, pages 167–182, Oxford, England, June 1986. Reprinted in G. Rozenberg, ed., Advances on Petri Nets '87, LNCS 266, pp.146–161, Springer Verlag, 1987.
19. M. Ajmone Marsan, G. Chiola, and A. Fumagalli. Timed Petri net model for accurate performance analysis of CSMA/CD bus LAN. *Computer Communications*, 10(6):304–312, December 1987.
20. M. Ajmone Marsan, G. Chiola, and G. Conte. Stochastic Petri nets as a tool for the analysis of high performance distributed architectures. In S. Kartashev and L. Kartashev, editors, *Supercomputing Systems.* Van Nostrand, 1990.
21. M. Ajmone Marsan, S. Donatelli, and F. Neri. GSPN models of Markovian multiserver multiqueue systems. *Performance Evaluation*, 11(4):227–240, 1990.
22. I.F. Akyildiz, G. Chiola, D. Kofman, and H. Körezlioğlu. Stochastic Petri nets modeling of the FDDI network protocol. In *Proc. of 11th Intern. Symp. on Protocol Specification, Testing, and Verification*, Stockholm, Sweden, June 1991.
23. H. Alla and P. Ladet. Coloured Petri nets : A tool for modeling, validation and simulation of FMS. In A. Kusiak, editor, *Flexible Manufacturing Systems : Methods and Studies.* Elsevier, 1986.
24. H.H. Ammar and R. W. Liu. Analysis of the generalized stochastic Petri nets by state aggregation. In *Proc. International Workshop on Timed Petri Nets*, pages 88–95, Torino, Italy, July 1985.
25. H.H. Ammar and S. Deng. Time warp simulation of stochastic Petri nets. In *Proc. 4th Intern. Workshop on Petri Nets and Performance Models*, pages 186–195, Melbourne, Australia, December 1991. IEEE-CS Press.
26. H.H. Ammar, S.M.R. Islam, and S. Deng. Performability analysis of parallel and distributed algorithms. In *Proc. 3rd Intern. Workshop on Petri Nets and Performance Models*, Kyoto, Japan, December 1989. IEEE-CS Press.
27. C. Anglano, S. Donatelli, and R. Gaeta. Parallel architectures with regular structure: a case study in modelling using SWN. In *Proc. 5th Intern. Workshop on Petri Nets and Performance Models*, Toulouse, France, October 1993. IEEE-CS Press.
28. F. Baccelli, M. Canales. Parallel Simulation of Stochastic Petri Nets using Recurrence Equations. ACM Tomacs, 1994.
29. F. Baccelli and B. Gaujal. Stationary regime and stability of free-choice Petri nets. In *Proceedings of the 11-th International Conference on Anal-*

ysis and Optimisation of Systems, Juan les Pins, France, June 1994. Springer Verlag.

30. F. Baccelli and P. Konstantopoulos. Estimates of cycle times in stochastic Petri nets. *Lecture Notes in Control and Information Sciences, Springer Verlag*, 177:1–21, 1992.
31. F. Baccelli and Z. Liu. Comparison properties of stochastic decision free Petri nets. *IEEE Transactions on Automatic Control*, 37:1905–1920, 1992.
32. F. Baccelli, G. Cohen, G.J. Olsder, and J.P. Quadrat. *Synchronization and Linearity: An Algebra for Discrete Event Systems.* Wiley, 1992.
33. F. Baccelli, G. Cohen, and B. Gaujal. Recursive equations and basic properties of timed Petri nets. *Jounal of Discrete Event Systems*, 1:415–439, 1992.
34. F. Baccelli, N. Bambos, and J. Walrand. Flows in stochastic marked graphs. *Probability in the Engineering and Informational Sciences*, 5:145–157, 1991.
35. F. Baccelli. Ergodic theory of stochastic Petri nets. *The Annals of Probability*, 20(1):375–396, 1992.
36. G. Balbo, S. Bruell , and M. Sereno. Arrival theorems for product-form stochastic Petri nets. In *Proc. 1993 SIGMETRICS Conference*, Nashville, Tennessee, USA, May 1993. ACM.
37. G. Balbo and G. Chiola. Stochastic Petri net simulation. In *Proc. 1989 Winter Simulation Conference*, Washington D.C., December 1989.
38. G. Balbo, G. Chiola, G. Franceschinis, and G. Molinar Roet. Generalized stochastic Petri nets for the performance evaluation of FMS. In *Proc. 1987 International Conference on Robotics and Automation*, pages 1013–1018, Raleigh, USA, April 1987. IEEE.
39. G. Balbo, G. Chiola, G. Franceschinis, and G. Molinar Roet. On the efficient construction of the tangible reachability graph of generalized stochastic Petri nets. In *Proc. Int. Workshop on Petri Nets and Performance Models*, Madison, WI, USA, August 1987. IEEE-CS Press.
40. G. Balbo, G. Chiola, S.C. Bruell, and P. Chen. An example of modelling and evaluation of a concurrent program using coloured stochastic Petri nets: Lamport's fast mutual exclusion algorithm. *IEEE Transactions on Parallel and Distributed Systems*, 3(2):221–240, March 1992.
41. G. Balbo, S. C. Bruell, and S. Ghanta. Combining queueing network and generalized stochastic Petri net models for the analysis of a software blocking phenomenon. In *Proc. International Workshop on Timed Petri Nets*, pages 208–225, Torino,Italy, July 1985.
42. G. Balbo, S. C. Bruell, and S. Ghanta. Combining queueing network and generalized stochastic Petri net models for the analysis of some software blocking phenomena. *IEEE Transactions on Software Engineering*, 12(4):561–576, April 1986.
43. G. Balbo, S. C. Bruell, and S. Ghanta. Combining queueing network and generalized stochastic Petri nets for the solution of complex models of system behavior. *IEEE Transactions on Computers*, 37(10):1251–1268, October 1988.

44. G. Balbo, S. Donatelli, G. Franceschinis, A. Mazzeo, N. Mazzocca, and M. Ribaudo. On the computation of performance characteristics of concurrent programs using GSPNs. *Performance Evaluation*, 19(2-3):195–222, March 1994.
45. G. Balbo, S. Donatelli, and G. Franceschinis. Understanding parallel programs behaviour through Petri net models. *Journal of Parallel and Distributed Computing*, 15(3), July 1992.
46. F. Baskett, K.M. Chandy, R.R. Muntz, and F.G. Palacios. Open, closed and mixed networks of queues with different classes of customers. *J. A. C. M.*, 22, 248-260, 1975.
47. M. Becker, C. Dekoninck, J.P. Prost, and B. Verrier. A stochastic Petri net model for the FPS/764. *Computer System Science and Engineering*, 1989.
48. A. Bertoni and M. Torelli. Probabilistic Petri nets and semi-Markov processes. In *Proc.* 2^{nd} *European workshop on Petri nets*, Bad Honnef, West Germany, September 1981.
49. A. Bobbio and A. Cumani. Discrete state stochastic systems with phase type distributed transition times. In *Proc. AMSE Int. Conf. on Modelling and Simulation*, Athens, 1984.
50. R.J. Boucherie. A characterisation of independence for competing Markov chains with applications to stochastic Petri nets. Proceedings of PNPM, 1993.
51. R.J. Boucherie and M. Sereno. On the traffic equations for batch routing queueing networks and stochastic Petri nets. 1994, in preparation.
52. R.J. Boucherie and M. Sereno. Product forms for stochastic Petri nets. 1994, in preparation.
53. R.J. Boucherie and N.M. Van Dijk. Product forms for queueing networks with state dependent multiple job transitions. *Advances in Applied Probability* 23, 152-187, 1991.
54. P. Buchholz. Aggregation and reduction techniques for hierarchical GCSPNs. In *Proceedings of the 5th International Workshop on Petri Nets and Performance Models*, pages 216–225, Toulouse, France, October 1993. IEEE-Computer Society Press.
55. P. Buchholz. Hierarchies in colored GSPNs. In M. Ajmone Marsan, editor, *Application and Theory of Petri Nets 1993*, volume 691 of *LNCS*, pages 106–125. Springer-Verlag, Berlin, 1993.
56. P. Buchholz. Numerical solution methods based on structured descriptions of Markovian models. In *Proc.* 5^{th} *Int. Conf. Modeling Techniques and Tools for Computer Performance Evaluation*, Torino, Italy, February 1991.
57. J. Campos and M. Silva. Embedded product-form queueing networks and the improvement of performance bounds for Petri net systems. *Performance Evaluation*, 18(1):3–19, July 1993.
58. J. Campos and M. Silva. Steady-state performance evaluation of totally open systems of Markovian sequential processes. In M. Cosnard and C. Girault, editors, *Decentralized Systems*, pages 427–438. Elsevier Science Publishers B.V. (North-Holland), Amsterdam, The Netherlands, 1990.

59. J. Campos and M. Silva. Structural techniques and performance bounds of stochastic Petri net models. In G. Rozenberg, editor, *Advances in Petri Nets 1992*, volume 609 of *LNCS*, pages 352–391. Springer-Verlag, Berlin, 1992.
60. J. Campos, B. Sánchez, and M. Silva. Throughput lower bounds for Markovian Petri nets: Transformation techniques. In *Proceedings of the 4rd International Workshop on Petri Nets and Performance Models*, pages 322–331, Melbourne, Australia, December 1991. IEEE-Computer Society Press.
61. J. Campos, F. Plo, and M. San Miguel. Boundedness on stochastic Petri nets. *Revista Matemática de la Universidad Complutense de Madrid*, 6(1):123–136, 1993.
62. J. Campos, G. Chiola, J. M. Colom, and M. Silva. Properties and performance bounds for timed marked graphs. *IEEE Transactions on Circuits and Systems—I: Fundamental Theory and Applications*, 39(5):386–401, May 1992.
63. J. Campos, G. Chiola, and M. Silva. Ergodicity and throughput bounds of Petri nets with unique consistent firing count vector. *IEEE Transactions on Software Engineering*, 17(2):117–125, February 1991.
64. J. Campos, G. Chiola, and M. Silva. Properties and performance bounds for closed free choice synchronized monoclass queueing networks. *IEEE Transactions on Automatic Control*, 36(12):1368–1382, December 1991.
65. J. Campos, J. M. Colom, H. Jungnitz, and M. Silva. A general iterative technique for approximate throughput computation of stochastic marked graphs. In *Proceedings of the 5th International Workshop on Petri Nets and Performance Models*, pages 138–147, Toulouse, France, October 1993. IEEE-Computer Society Press.
66. J. Campos, J. M. Colom, and M. Silva. Improving throughput upper bounds for net based models of manufacturing systems. In J. C. Gentina and S. G. Tzafestas, editors, *Robotics and Flexible Manufacturing Systems*, pages 281–294. North-Holland, Amsterdam, The Netherlands, 1992.
67. M. Canales. Simulation Parallèle de Réseaux de Petri Stochastiques. Thèse de Doctorat, Université de Nice, February 1994.
68. J.A. Carrasco. Automated construction of compound Markov chains from generalized stochastic high-level Petri nets. In *Proc. 3rd Intern. Workshop on Petri Nets and Performance Models*, pages 93–102, Kyoto, Japan, December 1989. IEEE-CS Press.
69. S. Caselli and G. Conte. GSPN models of concurrent architectures with mesh topology. In *Proc. 4th Intern. Workshop on Petri Nets and Performance Models*, pages 280–289, Melbourne, Australia, December 1991. IEEE-CS Press.
70. S. Caselli, G. Conte, F. Bonardi, and M. Fontanesi. Experiences on SIMD massively parallel GSPN analysis. In *Proc. 7^{th} Int. Conference on Modeling Techniques and Tools for Computer Performance Evaluation*, Vienna, Austria, May 1994.
71. S. Caselli, G. Conte, and U. Malavolta. Topology and process interaction in

concurrent architectures: a GSPN modeling approach. *Journal of Parallel and Distributed Computing*, 15(3):270–281, July 1992. Special Issue on Petri Net Modelling of Parallel Computers, guest editor's introduction.

72. K.M. Chandy and J. Misra. Distributed simulation: A case study in design and verification of distributed programs. *IEEE Transactions on Software Engineering*, 5(11):440–452, September 1979.
73. P. Chen, S.C. Bruell, and G. Balbo. Alternative methods for incorporating non-exponential distributions into stochastic Petri nets. In *Proc. 3rd Intern. Workshop on Petri Nets and Performance Models*, pages 187–197, Kyoto, Japan, December 1989. IEEE-CS Press.
74. G. Chiola and A. Ferscha. Distributed simulation of timed Petri nets: exploiting the net structure to obtain efficiency. In 14^{th} *International Conference on Application and Theory of Petri Nets*, Chicago, Illinois, June 1993. Springer Verlag.
75. G. Chiola and A. Ferscha. Distributed simulation of Petri nets. *Parallel and Distributed Technology*, 1(3):33–50, August 1993.
76. G. Chiola and A. Ferscha. Exploiting Petri net model structure to improve distributed simulation. In 26^{th} *Hawaii International Computer Science Symposium*, Honolulu, Hawaii, January 1993.
77. G. Chiola and A. Ferscha. Performance comparable design of synchronization protocols for distributed simulation. Technical report, Austrian Center for Parallel Computation, Technical University of Vienna, 1993.
78. G. Chiola and A. Ferscha. Self adaptive logical processes: the probabilistic distributed simulation protocol. In *Proc.* 27^{th} *Annual Simulation Symposium*, La Jolla, California, April 1994.
79. G. Chiola and R. Gaeta. Efficient simulation of parallel architectures exploiting symmetric well-formed Petri net models. In *SCS Western Simulation Multiconference '93*, pages 285–290, San Diego, California, January 1993. Society for Computer Simulation.
80. G. Chiola, C. Anglano, J. Campos, J. M. Colom, and M. Silva. Operational analysis of timed Petri nets and application to the computation of performance bounds. In *Proceedings of the 5th International Workshop on Petri Nets and Performance Models*, pages 128–137, Toulouse, France, October 1993. IEEE-Computer Society Press.
81. G. Chiola, C. Dutheillet, G. Franceschinis, and S. Haddad. Stochastic well-formed coloured nets and multiprocessor modelling applications. In K. Jensen and G. Rozenberg, editors, *High-Level Petri Nets. Theory and Application.* Springer Verlag, 1991.
82. G. Chiola, C. Dutheillet, G. Franceschinis, and S. Haddad. Stochastic well-formed coloured nets for symmetric modelling applications. *IEEE Transactions on Computers*, 42(11):1343–1360, November 1993.
83. G. Chiola, G. Bruno, and T. Demaria. Introducing a color formalism into generalized stochastic Petri nets. In *Proc.* 9^{th} *Europ. Workshop on Application and Theory of Petri Nets*, Venezia, Italy, June 1988.
84. G. Chiola, G. Franceschinis, and R. Gaeta. A symbolic simulation mechanism for well-formed coloured Petri nets. In *Proc. 25th SCS Annual*

Simulation Symposium, Orlando, Florida, April 1992.

85. G. Chiola, G. Franceschinis, and R. Gaeta. Modelling symmetric computer architectures by stochastic well-formed coloured nets. In 15^{th} *International Conference on Application and Theory of Petri Nets*, Zaragoza, Spain, June 1994.
86. G. Chiola, M. Ajmone Marsan, G. Balbo, and G. Conte. Generalized stochastic Petri nets: A definition at the net level and its implications. *IEEE Transactions on Software Engineering*, 19(2):89–107, February 1993.
87. G. Chiola, R. Gaeta, and M. Ribaudo. Designing an efficient tool for Stochastic Well-Formed Coloured Petri Nets. In R. Pooley and J. Hillston, editors, *Proc.* 6^{th} *Int. Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, pages 391–395, Edinburg, UK, September 1992. Antony Rowe Ltd.
88. G. Chiola, R. Gaeta, and M. Sereno. A simulation model of a double ring protocol based on timed well-formed coloured Petri nets. In *Intern. Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pages 259–264, San Diego, California, January 1993. Society for Computer Simulation.
89. G. Chiola. A graphical Petri net tool for performance analysis. In *Proc.* 3^{rd} *Int. Workshop on Modeling Techniques and Performance Evaluation*, Paris, France, March 1987. AFCET.
90. G. Chiola. A software package for the analysis of generalized stochastic Petri net models. In *Proc. Int. Workshop on Timed Petri Nets*, Torino, Italy, July 1985. IEEE-CS Press.
91. G. Chiola, M. Ajmone Marsan, and G. Balbo. Product-form solution techniques for the performance analysis of multiple-bus multiprocessor systems with nonuniform memory references. *IEEE Transactions on Computers*, 37(5):532–540, May 1988.
92. G. Chiola. Compiling techniques for the analysis of stochastic Petri nets. In R. Puigjaner and D. Potier, editors, *Proc.* 4^{th} *Int. Conf. Modeling Techniques and Tools for Computer Performance Evaluation*, Palma de Mallorca, Spain, September 1988. Plenum Press, New York.
93. G. Chiola. Simulation framework for timed and stochastic Petri nets. *International Journal in Computer Simulation*, 1(2):153–168, 1991.
94. G. Chiola. GreatSPN 1.5 software architecture. In *Proc.* 5^{th} *Int. Conf. Modeling Techniques and Tools for Computer Performance Evaluation*, Torino, Italy, February 1991.
95. G. Ciardo and K. S. Trivedi. A decomposition approach for stochastic reward net models. *Performance Evaluation*, 18(1):37–59, July 1993.
96. G. Ciardo, J. Muppala, and K.S. Trivedi. On the solution of GSPN reward models. *Performance Evaluation*, 12(4):237–253, July 1991.
97. G. Ciardo, J. Muppala, and K.S. Trivedi. SPNP: Stochastic Petri net package. In *Proc. 3rd Intern. Workshop on Petri Nets and Performance Models*, Kyoto, Japan, December 1989. IEEE-CS Press.
98. G. Ciardo, R. German, and C. Lindemann. A characterization of the stochastic process underlying a stochastic Petri net. In *Proc. 5th Intern.*

Workshop on Petri Nets and Performance Models, Toulouse, France, October 1993. IEEE-CS Press.

99. G. Cohen, D. Dubois, J.P. Quadrat, and M. Viot. A linear-system-theoretic view of discrete-event processes. In *Proceedings of the 22-nd Conference on Decision and Control.* IEEE, 1983.
100. J.L. Coleman. Algorithms for product form stochastic Petri nets - a new approach. In *Proceedings of the 5th International Workshop on Petri Nets and Performance Models*, pages 108-116, Toulouse, France, October 1993. IEEE-Computer Society Press.
101. J.L. Coleman, W. Henderson, and P.G. Taylor. Product form equilibrium distributions and a convolution algorithm for stochastic Petri nets, *Research Report, University of Adelaide*, 1992.
102. A. Cumani. Esp - a package for the evaluation of stochastic Petri nets with phase-type distributed transition times. In *Proc. Int. Workshop on Timed Petri Nets*, Torino, Italy, July 1985. IEEE-CS Press.
103. F. DeCindio and O. Botti. Comparing Occam2 program placements by a GSPN model. In *Proc. 4th Intern. Workshop on Petri Nets and Performance Models*, pages 216–221, Melbourne, Australia, December 1991. IEEE-CS Press.
104. A. Desrochers, H. Jungnitz, and M. Silva. An approximation method for the performance analysis of manufacturing systems based on GSPNs. In *Proceedings of the Rensselaer's Third International Conference on Computer Integrated Manufacturing*, pages 46–55, Rensselaer Polytechnic Institute, Troy, NY, USA, May 1992. IEEE-Computer Society Press.
105. M. Diaz. Modeling and analysis of communication and cooperation protocols using Petri net based models. *Computer Networks*, 6:419–441, December 1982.
106. S. Donatelli, and M. Sereno. On the product form solution for stochastic Petri nets. *Proceedings of the 13th international conference on application and theory of Petri nets*, Sheffield, UK, 154-172, 1992.
107. S. Donatelli. Superposed generalized stochastic Petri nets: definition and efficient solution. In *Proc.* 15^{th} *International Conference on Application and Theory of Petri Nets*, Zaragoza, Spain, June 1994.
108. S. Donatelli. Superposed stochastic automata: a class of stochastic Petri nets amenable to parallel solution. In *Proc. 4th Intern. Workshop on Petri Nets and Performance Models*, pages 54–63, Melbourne, Australia, December 1991. IEEE-CS Press.
109. J. B. Dugan and G. Ciardo. Stochastic Petri net analysis of a replicated file system. *IEEE Transactions on Software Engeneering*, 15(4):394–401, 1989.
110. J.B. Dugan, A. Bobbio, G. Ciardo, and K.S. Trivedi. The design of a unified package for the solution of stochastic Petri net models. In *Proc. Int. Workshop on Timed Petri Nets*, Torino, Italy, July 1985. IEEE-CS Press.
111. J.B. Dugan, K.S. Trivedi, R.M. Geist, and V.F. Nicola. Extended stochastic Petri nets: Applications and analysis. In *Proc. Performance '84*, Paris,

France, December 1984.

112. C. Dutheillet and S. Haddad. Aggregation and disaggregation of states in colored stochastic Petri nets: Application to a multiprocessor architecture. In *Proc. 3rd Intern. Workshop on Petri Nets and Performance Models*, Kyoto, Japan, December 1989. IEEE-CS Press.
113. C. Dutheillet and S. Haddad. Regular stochastic Petri nets. In *Proc. 10th Intern. Conf. Application and Theory of Petri Nets*, Bonn, Germany, June 1989.
114. P. Ferrara, A. Ferscha, I. Graf, and W. Müllner. Distributed timed Petri net simulation. Technical report, Dept. Statistics and Computer Science, University of Vienna, Lenaugasse 2/8, A-1080 Vienna, Austria, 1991.
115. A. Ferscha and G. Chiola. Accelerating the evaluation of parallel program performance models using distributed simulation. In *Proc. 7^{th} Int. Conference on Modeling Techniques and Tools for Computer Performance Evaluation*, Vienna, Austria, May 1994.
116. A. Ferscha. A Petri net approach for performance oriented parallel program design. *Journal of Parallel and Distributed Computing*, 15(3):188–206, July 1992. Special Issue on Petri Net Modelling of Parallel Computers.
117. G. Florin and S. Natkin. Les reseaux de Petri stochastiques. *Technique et Science Informatiques*, 4(1), February 1985.
118. G. Florin and S. Natkin. Matrix product form solution for closed synchronized queueing networks. In *Proc. 3rd Intern. Workshop on Petri Nets and Performance Models*, pages 29–39, Kyoto, Japan, December 1989. IEEE-CS Press.
119. G. Florin and S. Natkin. Necessary and sufficient ergodicity condition for open synchronized queueing networks. *IEEE Transactions on Software Engineering*, 15(4):367–380, April 1989.
120. G. Florin and S. Natkin. On open synchronized queuing networks. In *Proc. Int. Workshop on Timed Petri Nets*, Torino, Italy, July 1985. IEEE-CS Press.
121. G. Florin and S. Natkin. One-place unbounded stochastic Petri nets: Ergodicity criteria and steady-state solutions. *Journal of Systems and Software*, 6(1,2):103–115, May 1986.
122. G. Florin and S. Natkin. Rdps: a software package for the evaluation and the validation of dependable computer systems. In *Proc. SAFECOMP86*, Sarlat, France, 1986.
123. G. Franceschinis, R.R. Muntz. Computing bounds for the performance indices of quasi-lumpable Stochastic Well-Formed Nets. International Workshop on Petri Nets and Performance Models 1993, Toulouse, France, October 1993.
124. D. Frosch, and K. Natarajan. Product form solutions for closed synchronized systems of stochastic sequential processes. *Proceedings of 1992 International Computer Symposium*, Taichung, Taiwan, 392-402, December 13-15, 1992.
125. D. Frosch-Wilke. Exact performance analysis of a class of product form stochastic Petri nets. *Proceedings of the 1993 UK Performance Engineer-*

ing Workshop for Computer and Telecommunication Systems, Loughborough, UK, July 1993.

126. D. Frosch. Product form solutions for closed synchronized systems of stochastic sequential processes, Forschungsbericht Nr. 92-13, Universität Trier, Mathematik/Informatik, 1992.
127. R.M. Fujimoto. Parallel discrete event simulation. *Communications of the ACM*, 33(10):30–53, October 1990.
128. W.J. Gordon, and G.F. Newell. Closed queueing systems with exponential servers. *Operations Research*, 15, 254-265, 1967.
129. R. Gouet, F. Plo, M. San Miguel, and B. Ycart. Saturation conditions for stochastic Petri nets. In G. Rozenberg, editor, *Advances in Petri Nets 1993*, volume 674 of *LNCS*, pages 267–280. Springer-Verlag, Berlin, 1993.
130. P.J. Haas and G.S. Shedler. Regenerative simulation of stochastic Petri nets. In *Proc. Int. Workshop on Timed Petri Nets*, Torino, Italy, July 1985. IEEE-CS Press.
131. P.J. Haas and G. S. Shedler. Regenerative stochastic Petri nets. *Performance Evaluation*, 6(3):189–204, September 1986.
132. W. Henderson and D. Lucic. Aggregation and disaggregation through insensitivity in stochastic Petri nets. *Performance Evaluation*, 17, 91–114, 1993.
133. W. Henderson, C.E.M. Pearce, P.G. Taylor, and N.M. Van Dijk. Closed queueing networks with batch services. *Queueing Systems*, 6, 59-70, 1990.
134. W. Henderson, D. Lucic, and P.G. Taylor. A net level performance analysis of stochastic Petri nets. *Journal of the Australian Mathematical Society Series B*, 31, 176-187, 1989.
135. W. Henderson, and P.G. Taylor. Aggregation methods in exact performance analysis of stochastic Petri nets. *Proceedings of PNPM'89*, 12–18, Kyoto, Japan, December 1989.
136. W. Henderson, and P.G. Taylor. Open networks of queues with batch arrivals and batch services. *Queueing Systems*, 6, 71-88, 1990.
137. W. Henderson, and P.G. Taylor. Embedded processes in stochastic Petri nets. *IEEE Transactions on Software Engineering*, 17, 108-116, 1991.
138. O.C. Ibe and K.S. Trivedi. Stochastic Petri net models of polling systems. *IEEE Journal on Selected Areas in Communications*, 8(9), December 1990.
139. O.C. Ibe, A. Sathaye, R.C. Howe, and K.S. Trivedi. Stochastic Petri nets modeling of VAXcluster system availability. In *Proc. 3^{rd} Intern. Workshop on Petri Nets and Performance Models*, Kyoto, Japan, December 1989. IEEE-CS Press.
140. S.M.R. Islam and H.H. Ammar. On bounds for token probabilities in a class of generalized stochastic Petri nets. In *Proceedings of the 3rd International Workshop on Petri Nets and Performance Models*, pages 221–227, Kyoto, Japan, December 1989. IEEE-Computer Society Press.
141. J.R. Jackson. Jobshop-like queueing systems. *Management Science*, 10, 131-142, 1963.
142. A. Jean-Marie. Analytical computation of Lyapunov exponents in stochastic event graphs. *In: These proceedings.*

143. D.A. Jefferson. Virtual time. *ACM Transactions on Programming Languages and Systems*, 7(3):404–425, July 1985.
144. H. Jungnitz and A. Desrochers. Flow equivalent nets for the performance analysis of flexible manufacturing systems. In *Proceedings of the IEEE Robotics and Automation Conference 1991*, pages 122–127, 1991.
145. H. Jungnitz, A. Desrochers, and M. Silva. Approximation techniques for stochastic macroplace/macrotransition nets. In *Proceedings of the QMIPS Workshop on Stochastic Petri Nets*, pages 118–146, Sophia Antipolis, France, November 1992. INRIA. To appear in *Flexible Manufacturing Systems.*
146. H. Jungnitz, B. Sánchez, and M. Silva. Approximate throughput computation of stochastic marked graphs. *Journal of Parallel and Distributed Computing*, 15:282–295, 1992.
147. F.P. Kelly. (1979) *Reversibility and stochastic networks.* Wiley.
148. G.I. Klas and U. G. Seidel. A subsystem identification algorithm for the approximate solution of large GSPN models. In G. Balbo and G. Serazzi, editors, *Computer Performance Evaluation: Modelling Techniques and Tools*, pages 269–285. Elsevier, Amsterdam, The Netherlands, 1992.
149. G.I. Klas. Hierarchical solution of generalized stochastic Petri nets by means of traffic processes. In K. Jensen, editor, *Application and Theory of Petri Nets 1992*, volume 616 of *LNCS*, pages 279–298. Springer-Verlag, Berlin, 1992.
150. G. Klas. Protocol optimization for a packet-switched bus in case of burst traffic by means of GSPN. In 14^{th} *International Conference on Application and Theory of Petri Nets*, Chicago, Illinois, June 1993. Springer Verlag.
151. P. Konstantopoulos and F. Baccelli. On the cut-off phenomenon in some queueing systems. *J. Appl. Prob.*, 28:683–694, 1991.
152. A.A. Lazar and T.G. Robertazzi. Markovian Petri net protocols with product form solution. *Performance Evaluation*, 12, 67-77, 1991.
153. A.A. Lazar and T.G. Robertazzi. Markovian Petri net protocols with product form solution. *In: Proceedings of IEEE Infocom'87*, San Francisco, CA, pp. 1054-1062, March 1987.
154. P. Legato, A. Bobbio, and L. Roberti. The effect of failures and repairs on multiple cell production lines. In *Proc.* 20^{th} *Int. Symposium on Automotive Technology and Automation*, Florence, Italy, May 1989.
155. Y. Li and C.M. Woodside. Product form stochastic Petri nets and Norton's aggregation. Research Report, Carleton University, Ottawa, Canada, 1989.
156. Y. Li and C. M. Woodside. Iterative decomposition and aggregation of stochastic marked graph Petri nets. In G. Rozenberg, editor, *Advances in Petri Nets 1993*, volume 674 of *LNCS*, pages 325–349. Springer-Verlag, Berlin, 1993.
157. Y. Li and C. M. Woodside. Performance Petri net analysis of communications protocol software by delay-equivalent aggregation. In *Proceedings of the 4th International Workshop on Petri Nets and Performance Models*, pages 64–73, Melbourne, Australia, December 1991. IEEE-Computer Society Press.

158. M. Li and N.D. Georganas. Parametric analysis of stochastic Petri nets. In *Fifth International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, pages 287–299, Torino, Italy, 1992.
159. Chuang Lin and Dan C. Marinescu. On stochastic high level Petri nets. In *Proc. Int. Workshop on Petri Nets and Performance Models*, Madison, WI, USA, August 1987. IEEE-CS Press.
160. Chuang Lin and Dan C. Marinescu. Stochastic high level Petri nets and applications. *IEEE Transactions on Computers*, 37(7):815–825, July 1988.
161. C. Lindemann. An improved numerical algorithm for calculating steady-state solutions of Deterministic and Stochastic Petri net models. In *Proc. 4th Intern. Workshop on Petri Nets and Performance Models*, pages 176–185, Melbourne, Australia, December 1991. IEEE-CS Press.
162. J. Mairesse. A graphical representation for matrices in the (Max,+) algebra. INRIA Report n° 2078, 1993.
163. J. Mairesse. Products of irreducible random matrices in the (Max,+) algebra - Part I. INRIA Report n° 1939, 1993.
164. B. Mazigh, J. Gresser, and F. Simon. GSPN modeling methods for performance and dependability evaluation of a real-life flexible manufacturing system. In *Proc. 5th Intern. Workshop on Petri Nets and Performance Models*, Toulouse, France, October 1993. IEEE-CS Press.
165. P.M. Merlin and D. J. Farber. Recoverability of communication protocols: Implications of a theoretical study. *IEEE Transactions on Communications*, 24(9):1036–1043, September 1976.
166. J.F. Meyer, A. Movaghar, and W. H. Sanders. Stochastic activity networks: Structure, behavior, and application. In *Proc. International Workshop on Timed Petri Nets*, pages 106–115, Torino,Italy, July 1985.
167. M.K. Molloy. A CAD tool for stochastic Petri nets. In *Proc. ACM-IEEE-CS Fall Joint Computer Conference*, pages 1082–1091, Dallas, Texas, November 1986.
168. M.K. Molloy. Performance analysis using stochastic Petri nets. *IEEE Transactions on Computers*, 31(9):913–917, September 1982.
169. M.K. Molloy. *On the Integration of Delay and Throughput Measures in Distributed Processing Models.* PhD thesis, UCLA, Los Angeles, CA, 1981.
170. S. Natkin. *Les Reseaux de Petri Stochastiques et leur Application a l'Evaluation des Systemes Informatiques.* PhD thesis, CNAM, Paris, France, 1980. These de Docteur Ingegneur.
171. D.M. Nicol and S. Roy. Parallel simulation of timed Petri nets. In B. Nelson, D. Kelton, and G. Clark, editors, *Proc. 1991 Winter Simulation Conference*, pages 574–583, 1991.
172. J.D. Noe and G. J. Nutt. Macro E-nets representation of parallel systems. *IEEE Transactions on Computers*, 31(9):718–727, August 1973.
173. G.J. Olsder, J.A.C. Resing, R.E. de Vries, M.S. Keane, and G. Hooghiemstra. Discrete event systems with stochastic processing times. *IEEE Transactions on Automatic Control*, AC-35:299–302, 1990.
174. S. Rajsbaum. Stochastic marked graphs. In *Proceedings of the 4th International Workshop on Petri Nets and Performance Models*, pages 95–101,

Melbourne, Australia, December 1991. IEEE-Computer Society Press. To appear in *Information Processing Letters* with the title 'Upper and Lower Bounds for Stochastic Marked Graphs'.

175. C. Ramchandani. Analysis of Asynchronous Concurrent Systems by Timed Petri Nets. PhD thesis, MIT, Cambridge, MA, 1974.
176. R.R. Razouk and C.V. Phelps. Performance analysis using timed Petri nets. In *Proc. Int. Conf. on Parallel Processing*, pages 126–129, August 1984.
177. W. Reisig. Deterministic buffer synchronization of sequential processes. *Acta Informatica*, 18, 117-134, 1982.
178. T.G. Robertazzi. Computer Networks and Systems: Queueing Theory and Performance Evaluation. Springer-Verlag, 1990.
179. W.H. Sanders and J. F. Meyer. METASAN: a performability evaluation tool based on stochastic activity networks. In *Proc. ACM/IEEE-CS Fall Joint Computer Conference*, November 1986.
180. M. Sereno and G. Balbo. Computational algorithms for product form solution stochastic Petri nets. Proceedings of PNPM, 1993.
181. M. Sereno and G. Balbo. Mean value analysis of stochastic Petri nets. Technical report, University of Torino, 1993. Submitted for publication.
182. R.F. Serfozo. Markovian network processes: congestion dependent routing and processing. *Queueing Systems*, 5, 5–36, 1989.
183. C.R.M. Sundaram and Y. Narahari. Analysis of dynamic load balancing strategies using a combination of stochastic Petri nets and queueing networks. In 14th *International Conference on Application and Theory of Petri Nets*, Chicago, Illinois, June 1993. Springer Verlag.
184. F.J.W. Symons. Introduction to numerical Petri nets, a general graphical model of concurrent processing systems. *Australian Telecommunications Research*, 14(1):28–33, January 1980.
185. G.S. Thomas and J. Zahorjan. Parallel simulation of performance Petri nets: Extending the domain of parallel simulation. In B. Nelson, D. Kelton, and G. Clark, editors, *Proc. 1991 Winter Simulation Conference*, 1991.
186. N.M. Van Dijk. Queueing Networks and Product Forms, a Systems Approach. Wiley, 1993.
187. N. Viswanadham and Y. Narahari. Performance Modeling of Automated Manufacturing Systems. Prentice Hall, 1992.
188. I.Y. Wang and T.G. Robertazzi. Service stage Petri net models with product form. *Queueing Systems*, 7, 355–374, 1991.
189. P. Whittle. Partial balance and insensitivity. *J. Appl. Prob.*, 22, 168-176, 1985.
190. C.Y. Wong, T.S. Dillon, and K.E. Forward. Timed places Petri nets with stochastic representation of place time. In *Proc. Int. Workshop on Timed Petri Nets*, Torino, Italy, July 1985. IEEE-CS Press.
191. A. Zenie. Colored stochastic Petri nets. In *Proc. Int. Workshop on Timed Petri Nets*, pages 262–271, Torino, Italy, July 1985. IEEE-CS Press.
192. W.M. Zuberek. Timed Petri nets and preliminary performance evaluation. In *Proc. 7th Annual Symposium on Computer Architecture*, pages 88–96, La Baule, France, May 1980.

Applying Spectral Expansion in Evaluating the Performance of Multiprocessor Systems*

Markus Ettl
IMMD VII
University of Erlangen-Nürnberg
91058 Erlangen, Germany
Isi Mitrani
Computing Laboratory
University of Newcastle
Newcastle upon Tyne NE1 7RU, England

Two examples of non-standard queueing models are studied by means of the spectral expansion solution method. In the first example, the method is applied to the analysis of a multiprocessor system with a randomly varying number of processors. The convergence of traditional multiprocessor systems with breakdowns and repairs to that model is also examined. In the second example, a computational algorithm is proposed for the exact analysis of multiprocessor systems with multiprogramming and complex task structure described by precedence constraints. That algorithm can be extended to give approximate solutions using lumpability in Markov chains.

1 A MULTIPROCESSOR QUEUE WITH RANDOMLY VARYING NUMBER OF PROCESSORS

The system under study is a Markovian queue with a random number of identical parallel processors. Jobs arrive in a Poisson stream at rate σ and join a single unbounded queue. The job processing times are i.i.d. random variables, distributed exponentially with parameter μ. The operative periods of the processors are i.i.d. random variables, distributed exponentially with parameter ξ. When a processor breaks down, it is discarded. New processors join the processor pool in a Poisson stream with rate γ. A service which is interrupted by a breakdown is resumed on the next available processor from the point of interruption; there are no switching overheads.

This multiprocessor system is conveniently represented by a two-dimensional Markov-process $X = \{(I(t), J(t)) : t > 0\}$. Thereby, $I(t)$ is the number of operative processors and $J(t)$ is the number of jobs in the system. Denote the steady-state distribution of that process by

---

*Supported by the European Grant BRA-QMIPS of CEC DG XIII.

$$p_{i,j} = \lim_{t\to\infty} P\left(I(t) = i, J(t) = j\right), \quad i,j \geq 0 . \tag{1}$$

Since the breakdowns and arrivals of processors do not depend on the number of jobs present, the steady-state marginal distribution of the number of operative processors is obtained from

$$(\gamma + i\xi)\, p_{i,\cdot} = \gamma p_{i-1,\cdot} \; + \; (i+1)\xi p_{i+1,\cdot} \, , \quad i \geq 0 \, , \tag{2}$$

where $p_{-1,\cdot} = 0$ by definition.
These equations always have a normalizeable solution, given by the Poisson distribution with parameter γ/ξ:

$$p_{i,\cdot} = \left(\frac{\gamma}{\xi}\right)^i \frac{1}{i!} e^{-\frac{\gamma}{\xi}} . \tag{3}$$

Hence, the processing capacity of the system, which is defined as the average number of operative processors, is equal to

$$E[I] = \frac{\gamma}{\xi} . \tag{4}$$

The ergodicity condition for the two-dimensional Markov process X is that the offered load must be less than the processing capacity:

$$\frac{\sigma}{\mu} < \frac{\gamma}{\xi}. \tag{5}$$

Numerical Solution by Spectral Expansion

In order to apply the spectral expansion method to the multiprocessor system as described above, the state space must be truncated in one dimension. This is an approximation to the original model and, in general, an approximation error is incurred. The truncation of the state space is carried out in the i-dimension of the underlying Markov process, which means that the number of processors is subject to an upper bound, I_0. That bound is chosen so that the total probability mass of the discarded states is less than a desired tolerance level, ϵ. In other words, I_0 is determined as the least integer which satisfies

$$\sum_{i=I_0+1}^{\infty} \left(\frac{\gamma}{\xi}\right)^i \frac{1}{i!} e^{-\frac{\gamma}{\xi}} < \epsilon . \tag{6}$$

This leaves a state space of the form $(0 \leq I \leq I_0) \times (0 \leq J)$, such that

$$\sum_{i=0}^{I_0} \sum_{j=0}^{\infty} p_{i,j} \geq 1 - \epsilon . \tag{7}$$

The spectral expansion solution provides the steady-state distribution $\{p_{i,j} : 0 \leq i \leq I_0, j \geq 0\}$ in terms of the eigenvalues and left eigenvectors of a matrix polynomial

$$Q(\lambda) = Q_0 + Q_1\lambda + Q_2\lambda^2 , \tag{8}$$

where Q_0, Q_1 and Q_2 are matrices of dimension $(I_0 + 1) \times (I_0 + 1)$. These matrices describe the state transition rates involving arrivals, availability of processors and departures, respectively. In the present case, Q_0 and Q_2 are diagonal, while Q_1 is tri-diagonal. For details of the solution method, see Mitrani and Mitra [4].

The effect of the tolerance level ϵ on the solution is illustrated in Tables 1 and 2. These show the first moment and the squared coefficient of variation of the queue length for different values of ϵ. The processing capacity of the system was chosen to take on values between 2 to 8 and the arrival rate σ was adjusted to give an average utilization of 0.5 in all experiments.

	$E[J]$ for Different Values of ϵ									
$E[I]$	1e−1	1e−2	1e−3	1e−4	1e−5	1e−6	1e−7	1e−8	1e−10	1e−12
2	10.293	10.138	10.128	10.128	10.128	10.128	10.128	10.128	10.128	10.128
4	7.282	7.183	7.170	7.168	7.168	7.168	7.168	7.168	7.168	
6	6.388	6.255	6.244	6.243	6.243	6.243	6.243	6.243		
8	6.226	6.158	6.150	6.150	6.150	6.150				

TABLE 1. Mean Queue Length $E[J]$ Versus Different Values of the Tolerance Level ϵ $(\xi = 0.05, \mu = 1.0, \sigma = 0.5\gamma/\xi)$.

	$C^2[J]$ for Different Values of ϵ									
$E[I]$	1e−1	1e−2	1e−3	1e−4	1e−5	1e−6	1e−7	1e−8	1e−10	1e−12
2	2.873	2.924	2.928	2.928	2.928	2.928	2.928	2.928	2.928	2.928
4	3.413	3.452	3.457	3.458	3.458	3.458	3.458	3.458	3.458	
6	2.816	2.835	2.836	2.837	2.837	2.837	2.837	2.837		
8	1.920	1.910	1.909	1.909	1.909	1.909				

TABLE 2. Squared Coefficient of Variation of Queue Length $C^2[J]$ Versus Different Values of the Tolerance Level ϵ $(\xi = 0.05, \mu = 1.0, \sigma = 0.5\gamma/\xi)$.

It is apparent from Tables 1 and 2 that a three-digit accuracy in both the mean and squared coefficient of variation of the queue length is reached within, say, a 1e−2 to e−4 tolerance level. Our computational experience from all experiments that we have carried out strongly supports the assumption that a tolerance of e−4 is sufficient for all practical modelling purposes.

Figure 1 displays the variation of I_0, which is required for truncating the state space, with the tolerance level ϵ. The average number of operative servers ranges from 2 to 10. The ϵ values are plotted on a log-scale. It is readily seen that increasing the tolerance level one order of magnitude is at a linear expense in the number of processors. That means, reducing the approximation error is obtained at low cost when considering the amount of additional states.

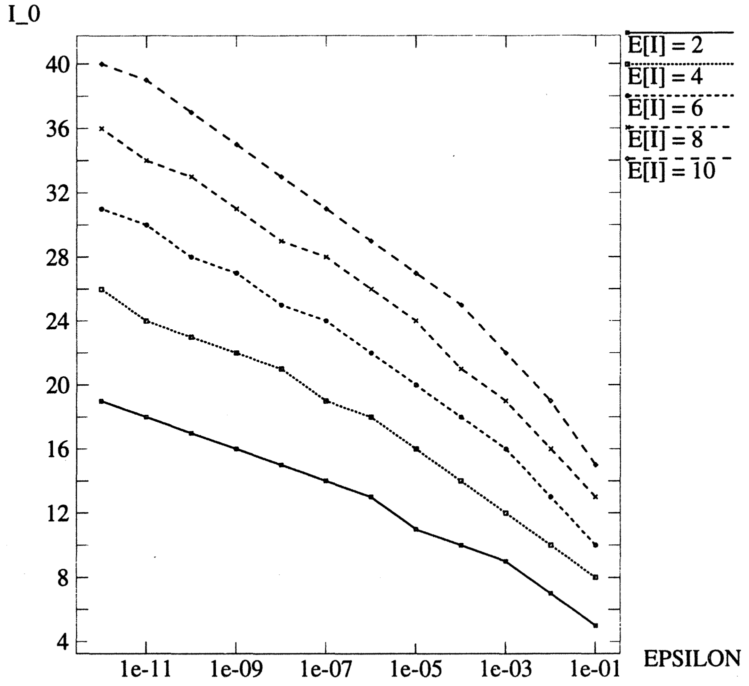


FIGURE 1. State Space Truncation Index I_0 Versus Different Values of Tolerance Level ϵ ($\xi = 0.05, \mu = 1.0, \sigma = 0.5\gamma/\xi$).

The Convergence of Finite Multiprocessor Queues

Consider an M/M/N queue where each processor breaks down and is repaired independently of the others, and where the operative and repair periods are distributed exponentially with means $1/\xi$ and $1/\eta$, respectively. Suppose that the number of processors and the mean repair period tend to infinity, so that their ratio has a finite limit, γ:

$$\gamma = \lim_{N\to\infty,\eta\to 0} N\eta \, . \tag{9}$$

Then it is intuitively obvious (but rather difficult to prove rigorously), that in the limit, this queue behaves like the one considered in the previous sections: permanent breakdowns with rate ξ, and Poisson arrivals of processors with rate γ. We wish to investigate this convergence numerically, using the exact spectral expansion solution of the M/M/N queue, together with the truncation method for the limiting system.

Recall that the average number of operative servers in an M/M/N multiprocessor system with breakdowns and repairs, $E[I_N]$, is given by

$$E[I_N] = N \frac{\eta}{\eta + \xi} \, . \tag{10}$$

Substituting formally η by γ/N and taking the limit in N, we find

$$\lim_{N\to\infty} E[I_N] = \lim_{N\to\infty} \frac{\gamma}{\frac{\gamma}{N}+\xi} = \frac{\gamma}{\xi}\,, \tag{11}$$

which is the average number of operative servers in the infinite system, given by equation (4). Note that, on average, the number of operative servers in the finite M/M/N systems is always less than those in the infinite one. Thus, the infinite system provides an upper bound on the performance of a sequence of finite multiprocessor systems.

Figures 2 to 4 show the speed of convergence of the finite systems, with γ/ξ equal to 2, when the offered load is increased. Each figure contains plots of the average number of operative servers, $E[I_N]$, and the mean queue length, $E[J_N]$. The average values of the infinite system are included for comparison and do not vary in N. Figures 5 to 7 are similar, with the ratio γ/ξ equal to 5. It is seen in all plots that the infinite system bounds the average queue length of the M/M/N breakdown-repair models when the number of processors N is increased. The speed of convergence depends heavily on the ratio of γ/ξ and the offered load. The speed of convergence decreases when γ/ξ and the offered load increase. This becomes even more evident when checking the numerical values in Tables 3 to 4, which are included for better reference.

M/M/N–Approximation for $E[I] = 2$

ρ	$E[J]$	$E[J_6]$	$E[J_8]$	$E[J_{10}]$	$E[J_{12}]$	$E[J_{14}]$	$E[J_{16}]$	$E[J_{18}]$	$E[J_{20}]$	$E[J_{22}]$	$E[J_{24}]$
0.25	2.31	3.03	2.85	2.74	2.67	2.61	2.58	2.55	2.52		
0.50	10.12	17.30	15.01	13.84	13.12	12.64	12.29	12.03	11.82	11.65	
0.75	44.58		193.64	119.74	94.96	82.52	75.02	70.18	66.42	63.71	61.61

TABLE 3. Mean Queue Length $E[J_N]$ Versus Increasing Values of N from Figures 2 to 4.

M/M/N–Approximation for $E[I] = 5$

ρ	$E[J]$	$E[J_{12}]$	$E[J_{14}]$	$E[J_{16}]$	$E[J_{18}]$	$E[J_{20}]$	$E[J_{22}]$	$E[J_{24}]$	$E[J_{26}]$	$E[J_{28}]$	$E[J_{30}]$
0.25	1.63	2.17	2.08	2.01	1.96	1.92	1.89	1.87			
0.50	6.55	15.77	13.72	12.42	11.51	10.85	10.34	9.93	9.61	9.34	9.12
0.75	34.75			771.65	281.36	183.15	140.98	117.51	102.55	92.18	84.57

TABLE 4. Mean Queue Length $E[J_N]$ Versus Increasing Values of N from Figures 5 to 7.

The computational effort of solving an M/M/N system by the spectral expansion method is on the order of $O(N^3)$, i.e., it increases quite rapidly with N. However, that effort does not increase with the offered load. On the other hand, we have experienced numerical problems associated with ill-conditioned matrices when N is greater than about 100.

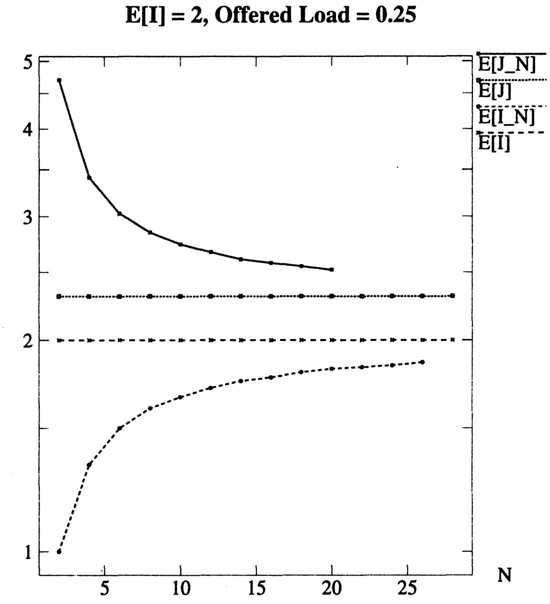


FIGURE 2. Mean Queue Length $E[J_N]$ and Mean Number of Operative Servers $E[I_N]$ Versus Increasing Values of N ($\xi = 0.05, \mu = 1.0, \gamma/\xi = 2, \sigma = 0.25\gamma/\xi$).

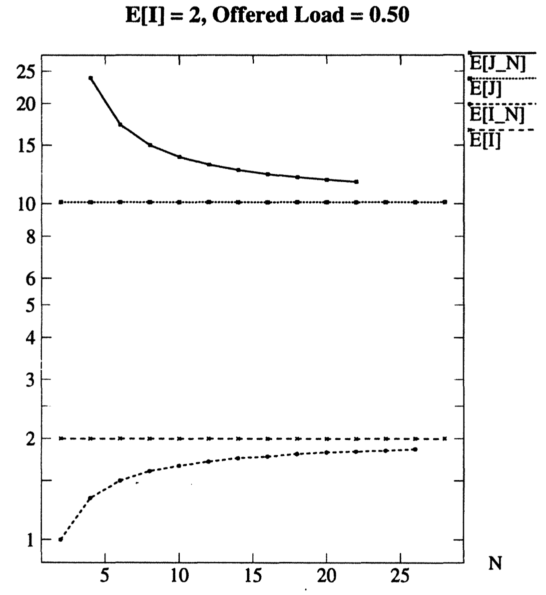


FIGURE 3. Mean Queue Length $E[J_N]$ and Mean Number of Operative Servers $E[I_N]$ Versus Increasing Values of N ($\xi = 0.05, \mu = 1.0, \gamma/\xi = 2, \sigma = 0.5\gamma/\xi$).

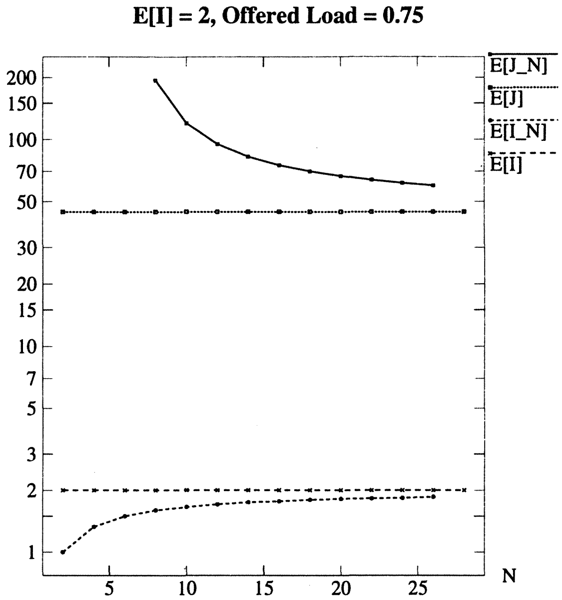


FIGURE 4. Mean Queue Length $E[J_N]$ and Mean Number of Operative Servers $E[I_N]$ Versus Increasing Values of N ($\xi = 0.05, \mu = 1.0, \gamma/\xi = 2, \sigma = 0.75\gamma/\xi$).

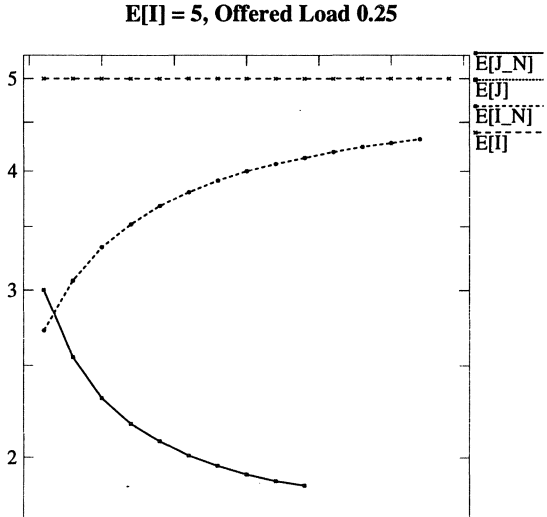


FIGURE 5. Mean Queue Length $E[J_N]$ and Mean Number of Operative Servers $E[I_N]$ Versus Increasing Values of N ($\xi = 0.05, \mu = 1.0, \gamma/\xi = 5, \sigma = 0.25\gamma/\xi$).

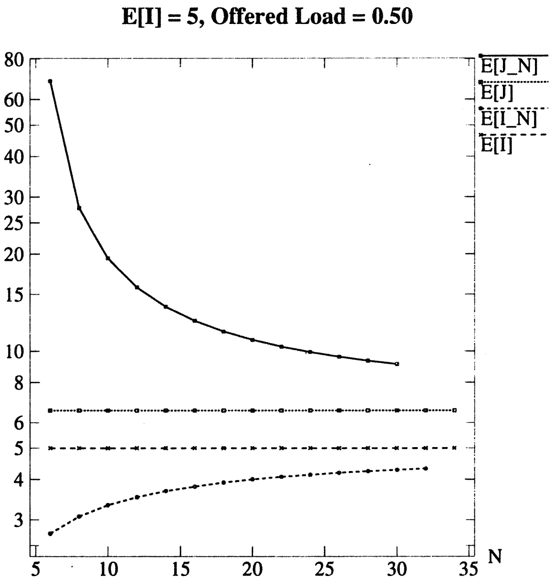


FIGURE 6. Mean Queue Length $E[J_N]$ and Mean Number of Operative Servers $E[I_N]$ Versus Increasing Values of N ($\xi = 0.05, \mu = 1.0, \gamma/\xi = 5, \sigma = 0.5\gamma/\xi$).

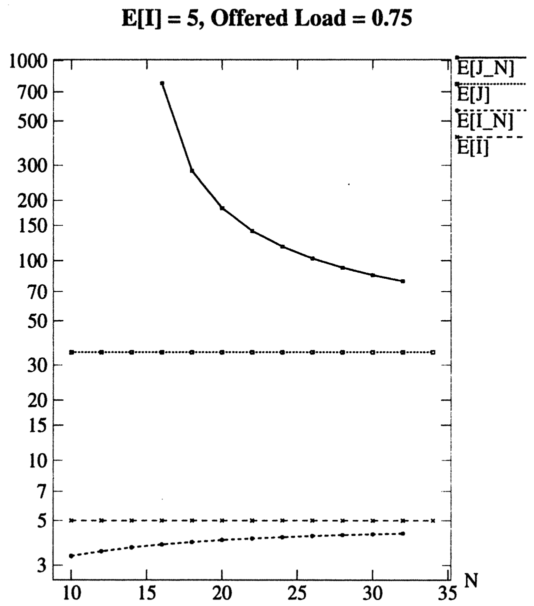


FIGURE 7. Mean Queue Length $E[J_N]$ and Mean Number of Operative Servers $E[I_N]$ Versus Increasing Values of N ($\xi = 0.05, \mu = 1.0, \gamma/\xi = 5, \sigma = 0.75\gamma/\xi$).

2 A MULTIPROCESSOR QUEUE WITH MULTIPROGRAMMING AND COMPLEX TASK STRUCTURE

In this section, we apply the spectral expansion method to the performance evaluation of a multiprocessor system with multiprogramming and complex task structure. In general, the system is characterized by the workload model, the multiprocessor type, the operating mode and the communication network:

Workload Model

- Jobs arrive to the system in a Poisson stream with parameter σ.
- Each job represents a concurrent program that consists of a finite set of tasks with different processing times.
- The precedence relationships between tasks are described by a series-parallel task graph (we assume that all jobs have the same task graph structure).

Multiprocessor Type

- N identical processors
- No breakdowns
- Global memory of unlimited size

Operating Mode

- The multiprocessor can simultaneously serve tasks of different jobs on a FCFS basis.

Communication Network

- No additional overhead is incurred for communication between processors.

It is assumed that the task graphs consist of one or more processing stages. A processing stage is either for synchronization (serial stage) or for processing of independent tasks (parallel stage). Figure 8 shows a 3-stage series parallel task graph with 1 parallel and 2 serial processing stages.

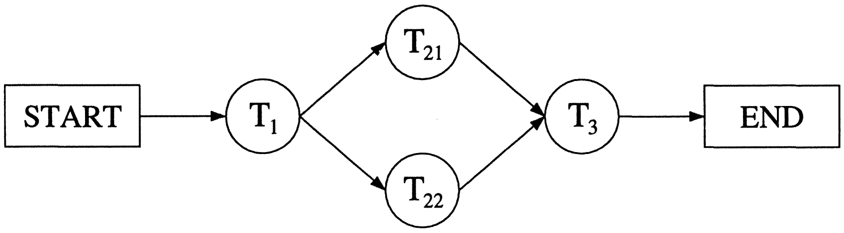


FIGURE 8. Series-Parallel Task Graph with 3 Processing Stages.

The processing times of each task are exponentially distributed random variables with different rates. The processing times of different tasks are assumed to be independent.

Numerical Solution by Spectral Expansion

In order to apply the spectral expansion method to the above multiprocessor queue with multiprogramming, we need to find a two-dimensional representation of the underlying Markov process, defined on a semi-infinite strip. Clearly, the infinite dimension is for the number of jobs in the system. As for the finite dimension, we must enumerate all possible states of the servers, which will be referred to as *operating states* for the rest of the paper. An operating state is defined by the number of jobs that are currently being processed (the degree of multiprogramming) and, for each job in operation, the set of tasks that are in service. As an example, the operating state $[2, (T_1, T_2)]$ indicates that there are two jobs being processed in parallel, one of which is in task T_1 and the other is in task T_2, respectively.

Example 1: M/M/2 with Serial Task Graphs

Suppose we have an M/M/2 multiprocessor system and a task graph consisting of three serial processing stages with processing rates μ_1 to μ_3. The operating states for that system, including the state transitions within one level, are shown in Figure 9.

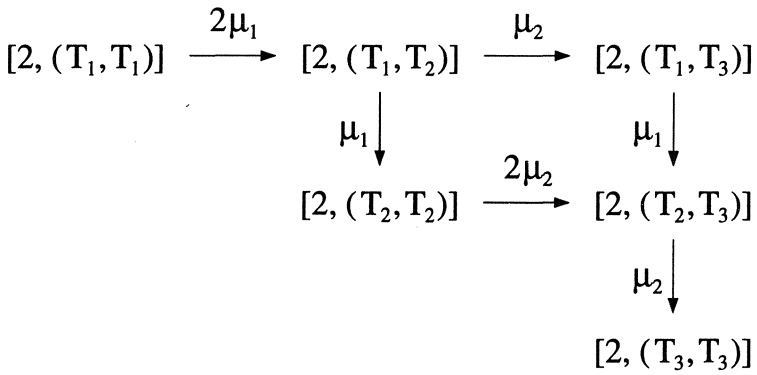


FIGURE 9. Operating States of an M/M/2 Multiprocessor System with a 3-Stage Serial Task Graph Structure.

Referring to the example from Figure 9, we can number the operating states in lexicographic order. It is then easily established that the matrix, A, of transition rates between different operative states for a given number of jobs ($j \geq 2$), is upper-triangular; the matrix, C, of transition rates associated with job departures is lower-triangular:

$$A = \begin{vmatrix} 0 & 2\mu_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu_2 & \mu_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu_1 & 0 \\ 0 & 0 & 0 & 0 & 2\mu_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu_2 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix} \qquad C = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2\mu_3 & 0 & 0 & 0 \end{vmatrix}$$

The matrix, B, of transition rates associated with job arrivals is diagonal, since the latter do not affect the operating state of the system:

$$B = \text{diag}(\sigma, \sigma, \dots, \sigma).$$

These matrices are used in the definition of the matrix polynomial (8), and hence in the implementation of the spectral expansion solution. Observe that the threshold beyond which the transition rates become independent of the number of jobs is equal, in general, to the number of processors N.

The construction of the transition rate matrices becomes more complicated when parallel processing stages are involved. In favor of a constructive system design, we propose to generate states systematically, rather than enumerating them in a brute-force approach. Thereby, we develop a method which allows to describe systems with complex task graph structures and build them systematically from smaller ones. The proposed solution method is exact.

The method consists of three basic steps and proceeds as follows: In the first step, each parallel processing stage is substituted by one serial stage in order to obtain a task graph with only serial stages (**reduction step**). Having constructed the reduced task graph, it is straightforward to find the set of operating states for the multiprocessor system with the reduced task graph structure. We call such operating states the "big" states. Notice that the matrices associated with such a reduced model can be constructed in much the same way as it was demonstrated in Example 1.

Once the big states have been established for the reduced model, we expand each big state into a set of "small" states by re-introducing the parallel processing stages (**expansion step**). The set of small states represents the state space of the original model. Using the small states, we can then construct the matrices for spectral expansion and invoke the spectral expansion solution package in order to calculate the performance metrics of the original model (**analysis step**).

To illustrate the solution method presented above, we consider an M/M/2 multiprocessor system with 3-stage series-parallel task graph input.

Example 2: M/M/2 with Series-Parallel Task Graphs

Assume we are given an M/M/2 multiprocessor system with jobs having the task graph structure shown in Figure 8. Within the reduction step, we obtain a task graph with three serial stages such that the big states are identical to those depicted in Figure 9. As an example, consider the big state $[2, (T_1, T_2)]$.

Carrying out the expansion step for that big state yields 4 small states since processing stage 2, in effect, is a parallel stage:

$[2,(T_1,(T_{21},0))]$: one job in task T_1, the other in task T_{21} with task T_{22} awaiting execution

$[2,(T_1,(T_{21},-))]$: one job in task T_1, the other in task T_{21} with task T_{22} finished

$[2,(T_1,(0,T_{22}))]$: one job in task T_1, the other in task T_{22} with task T_{21} awaiting execution

$[2,(T_1,(-,T_{22}))]$: one job in task T_1, the other in task T_{22} with task T_{21} finished

Obviously, the expansion of each big state into a set of small states, as well as the construction of transitions between the small states, can be obtained by direct enumeration.

The matrices A and C for Example 2 have the following shape:

$$A = \begin{vmatrix} A_{11} & A_{12} & 0 & 0 & 0 & 0 \\ 0 & A_{22} & A_{23} & A_{24} & 0 & 0 \\ 0 & 0 & A_{33} & 0 & A_{35} & 0 \\ 0 & 0 & 0 & A_{44} & A_{45} & 0 \\ 0 & 0 & 0 & 0 & A_{55} & A_{56} \\ 0 & 0 & 0 & 0 & 0 & A_{66} \end{vmatrix} \qquad C = \begin{vmatrix} C_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & C_{22} & 0 & 0 & 0 & 0 \\ C_{31} & 0 & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & C_{52} & 0 & 0 & C_{55} & 0 \\ 0 & 0 & C_{63} & 0 & 0 & C_{66} \end{vmatrix}$$

The matrix B is diagonal in appropriate dimensions. Note the similarity in structure to the matrices defined in Example 1 for the serial task graph model, the difference being that additional entries come to the diagonal which are due to state transitions within big states.

Computational Algorithm

In summary, we propose the following three-step computational algorithm for the exact numerical analysis of a multiprocessor system with multiprogramming and complex task structure:

Step 1 (Reduction Step):

- Substitute each parallel processing stage of the original task graph by a serial one.
- Construct the set of operating states (big states) for the multiprocessor system with the reduced task graph.

Step 2 (Expansion Step):

- Generate the set of small states from the big states by direct enumeration.
- Eliminate all redundant small states, i.e. those states that are not reached.

- Construct the matrices as required for the spectral expansion method. The structure of the matrices is defined by the reduced model.

Step 3 (Analysis Step):

- Call the spectral expansion package to calculate the performance metrics of the original model.

We note that the number of big states being constructed within the Reduction Step of the computational algorithm is equal to

$$M = \binom{N+S-1}{S-1}, \tag{12}$$

where N denotes the number of processors and S the number of processing stages of the original task graph, respectively.

The benefit of our approach is that states are generated in a systematic way. The global structure of the matrices A, B and C is defined by the reduced model, which means that a transition between any two small states is feasible only if there is a transition between the big states which they belong to in the reduced model.

In order to increase computational efficiency, the proposed method can be combined with state aggregation techniques for Markov chains. For instance, instead of expanding each big state, we can attempt to lump the set of small states associated with one big state into a single state. If the lumpability condition holds, the solution of the lumped Markov chain will be exact. If the lumpability condition holds within ϵ-accuracy, we can at least obtain bounds on the performance of the original model at low computational cost (Franceschinis and Muntz [2]). A survey on exact and approximate state aggregation techniques is given by Schweitzer [5] and in the book of Courtois [1].

The computational algorithm presented in this section can be applied, in principle, to multiprocessor systems with any number of processors. Furthermore, the algorithm can be adapted to handle multiprocessors with breakdown and repair, as well as more general task graph structures. We believe that great computational savings will be gained when analyzing large multiprocessor systems with concurrent programs, in particular when the basic algorithm is extended around state aggregation techniques.

CONCLUSIONS

The empirical studies reported here show that the spectral expansion solution method is a useful and efficient tool for performance evaluation. It applies to a large class of models whose state spaces are (or can be approximated by) two-dimensional semi-infinite strips. The computational complexity of the method compares very favourably with its main alternative, the Matrix-Geometric solution (some comparisons are reported in Mitrani and Chakka [3]).

There are bounds on the applicability of the spectral expansion method, imposed by the numerical properties of the algorithm (such bounds exist for all other solution methods). On present evidence, problems of size greater than 50-100 (as measured by the number of states in the finite dimension), are liable to exhibit symptoms of numerical instability. Extending that range would be a worthwhile topic of further research.

REFERENCES

1. Courtois, P.J. (1977). "Decomposability: Queueing and Computer Systems Applications". Academic Press, New York.
2. Franceschinis, G. and R.R. Muntz (1993). "Bounds for Quasi-Lumpable Markov Chains". In: Performance 93, G. Iazeolla and S. Lavenberg (eds.), 220-243.
3. Mitrani, I. and R. Chakka (1993). "Application and Evaluation of the Spectral Expansion Solution Method". Proceedings of the QMIPS Workshop on Formalisms, Principles and State-of-the-Art, Pommersfelden, 253-267.
4. Mitrani, I. and D. Mitra (1991). "Spectral Expansion Method for Random Walks on the Semi-Infinite Strip". IMACS Symposium on Iterative Methods in Linear Algebra, Brussels.
5. Schweitzer, P. (1991). "A Survey of Aggregation-Disaggregation in Large Markov Chains". In: Numerical Solutions of Markov Chains, Marcel Dekker Inc., 63-88.

The Compensation Approach Applied to a 2×2 Switch

O.J. Boxma*
CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
and
Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands

G.J. van Houtum
Eindhoven University of Technology,
Department of Mathematics and Computing Science,
P.O. Box 513, 5600 MB Eindhoven, The Netherlands

In this paper we analyze an asymmetric 2 × 2 buffered switch, fed by two independent Bernoulli input streams. We derive the joint equilibrium distribution of the numbers of messages waiting in the two output buffers. This joint distribution is presented explicitly, without the use of generating functions, in the form of a sum of two alternating series of product-form geometric distributions. The method used is the so-called compensation approach, developed by Adan, Wessels and Zijm.

1 INTRODUCTION

In parallel data processing networks, switching systems are being used to route messages and to resolve conflicts. The simplest basic unit in a switching system is the 2×2 clocked buffered switch. In this paper we study an asymmetric 2×2 clocked buffered switch under the simplest possible assumptions:

(i) all transfers are governed by a single discrete-time clock;

(ii) the two input ports receive messages according to two independent Bernoulli streams, viz., at each input port either 0 or 1 message arrives at each clock cycle;

(iii) the input ports route these messages to one of the two output ports according to certain routing probabilities.

---

*Supported by the European Grant BRA-QMIPS of CEC DG XIII

Under assumptions (ii) and (iii) the arrival processes at the two output ports are in general dependent, and hence the queue lengths at the two output buffers are coupled.

The 2×2 clocked buffered switch has recently been analyzed in detail by Jaffe [12], to whom we also refer for further references concerning buffered switches. Jaffe restricts his attention to the completely symmetric switch. He obtains the equilibrium joint queue length distribution at the two output buffers (as seen at the end of a clock cycle). He applies a uniformization technique that was used by Flatto and McKean [9] (see also Kingman [13]) for the shortest queue problem. Jaffe's results in [12] are presented in the form of a bivariate generating function.

The goal of the present paper is to show how the *asymmetric* 2×2 buffered switch can be analyzed completely, using the so-called compensation approach. The compensation approach has been developed in a series of papers by Adan et al. [2-4,6]; a detailed exposition of the method is presented in the Ph.D. thesis of Adan [1]. In its present form the method is limited in the sense that it has only been shown to work for two-dimensional random walks on the lattice of the first quadrant, with the following properties (see [6]):

(i) only transitions to neighbouring states occur;

(ii) no transitions from interior points are permitted to the North, North-East and East;

(iii) semi-homogeneity: the same transition rates occur for all interior points, and similarly for all points on the horizontal boundary and for all points on the vertical boundary.

But when these conditions are satisfied - as is the case in the model under consideration - then the method is very powerful; it yields an explicit expression, without transforms, for the steady-state distribution of the two-dimensional random walk. In fact we shall analyze a two-dimensional random walk that contains the random walk for the 2×2 switch as a special case. The random walk under consideration satisfies (i), (ii) and (iii) above, plus the

(iv) projection property: at each point of the horizontal boundary the transition rate to the West is the sum of the transition rates to the West and South-West in any point above it in the interior, and a similar property holds for the transition rate to the East, while the transition rates to the North-West, North and North-East are the same as the corresponding rates for the interior; similarly for the transition rates on the vertical boundary and in the origin.

The essence of the compensation approach is to characterize the class of *product-form* solutions satisfying the equilibrium equations at the interior points, and then to use the solutions in this class to construct a linear combination of product-form solutions which also satisfies the boundary conditions. The construction uses a compensation idea: after introducing the first product form

$\alpha^m \beta^n$ for the equilibrium distribution $\{p_{m,n}\}$ that satisfies the equilibrium equations for $m \geq 1$, $n \geq 1$, product-form terms are added so as to alternately compensate for errors on the two boundaries $m = 0$ and $n = 0$. The compensation approach has been successfully applied to the symmetric shortest queue problem [3], to the asymmetric shortest queue problem [4], and to a multiprogramming queue [2] originally studied by Hofri [10] (see also [5]). An attractive feature of the compensation approach is the relative simplicity of the resulting equilibrium distribution as a sum of product forms. The structure of the solution can easily be exploited for numerical analysis and leads to efficient algorithms, with the advantage of tight error bounds.

In queueing and random walk theory there is quite a sharp distinction between 'easy' and 'hard' two-dimensional problems. Some two-dimensional problems can be solved almost trivially, like the equilibrium queue length distribution in two M/M/1 queues in series. However, for most two-dimensional problems no exact solution is known. A few two-dimensional queueing problems have recently yielded to an exact analysis, that is usually based on complex function theory. An often studied example of the latter type is the shortest queue problem, that was originally solved by Kingman [13] and Flatto and McKean [9], and that was later analyzed (in the asymmetric case) using the boundary value method [8]. The boundary value method aims to translate the problem of determining the generating function of the joint distribution of the random walk/queueing problem into a boundary value problem of Dirichlet, Riemann or Riemann-Hilbert type. In [8] the same method is shown to solve (i) the M/G/2 queue, (ii) a model of two processors with coupled service speeds, and (iii) a polling model with two queues and 1-limited service at the queues. By now the boundary value method has established itself as a powerful method for a large class of two-dimensional random walks in the first quadrant.

We consider the 2×2 buffered switch problem as a particularly simple one in the class of 'hard' problems like the shortest queue problem. It is therefore most suitable for testing and comparing techniques like the compensation approach, the generating function approach with uniformization of Flatto and McKean, and the boundary value method. At the end of this paper we return to a discussion of these various methods for analyzing the buffered switch. Here it suffices to remark that the *symmetric* buffered switch has not only been analyzed by the method of Flatto and McKean (Jaffe [12]), but also by the boundary value method (Jaffe [11]). Those two methods will undoubtedly also work for the *asymmetric* switch studied in the present paper. Here we do not investigate this further, but we show that the 2×2 buffered switch is particularly well suited for an analysis by the compensation approach, enabling us to obtain a remarkably simple and elegant expression for the joint queue length distribution.

The organization of the paper is as follows. In section 2 we present the model in detail, and we derive the Kolmogorov equations for the equilibrium joint queue length distribution. This equilibrium distribution is derived in sections 3 and 4, using the compensation approach; in its application several simplifications arise which are due to the projection property. In section 3 we show that for the present model the compensation approach generates two alternating series of two-dimensional product-form (geometric) distributions, and in section 4 we prove that the joint queue length distribution $\{p_{m,n}\}$ is obtained by simply taking the sum of these two series:

$$p_{m,n} = \sum_{i=0}^{\infty}(1-\beta_i)\beta_i^n[(1-\alpha_i)\alpha_i^m - (1-\alpha_{i+1})\alpha_{i+1}^m]$$

$$+\sum_{i=0}^{\infty}(1-\tilde{\alpha}_i)\tilde{\alpha}_i^m(1-\tilde{\beta}_i)\tilde{\beta}_i^n - (1-\tilde{\beta}_{i+1})\tilde{\beta}_{i+1}^n], \quad m \geq 0, n \geq 0, m+n \geq 1. \quad (1)$$

The α_i, β_i, $\tilde{\alpha}_i$ and $\tilde{\beta}_i$ are specified in section 3 and the convergence of the above sums for all $m \geq 0, n \geq 0$ with $m+n \geq 1$ is discussed in section 4. Due to divergence of the above sums for $m = 0$ and $n = 0$, formula (1) does not hold for $p_{0,0}$. In section 5 we determine the coefficient of correlation of the two queue lengths for the symmetric case, and show it to be negative. Finally, section 6 is devoted to a comparison between the results and methods of Jaffe's papers [11,12] and the present one.

2 THE MODEL AND THE EQUILIBRIUM EQUATIONS

A 2 × 2 buffered switch is a switch with 2 input and 2 output ports. Such a switch is modeled as a discrete time queueing system with 2 parallel servers and 2 types of arriving jobs (see figure 1). Jobs of type i, $i = 1,2$, are assumed to arrive according to a Bernoulli stream with rate r_i, $0 < r_i \leq 1$. This means that every time unit (= clock cycle) the number of arriving jobs of type i is one with probability r_i and zero with probability $1 - r_i$. Jobs always arrive at the beginning of a time unit and once a job of type i has arrived, it joins the queue at server j with probability $t_{i,j}$, $t_{i,j} > 0$ for $j = 1,2$ and $t_{i,1} + t_{i,2} = 1$. Jobs that have arrived at the beginning of a time unit are immediately candidates for service. A server serves exactly one job per time unit, if one is present. For the sake of completeness the servers are assumed to have FCFS service discipline, but this does not influence the analysis. We assume that for each server the average number of arriving jobs per time unit is less than one, i.e.

$$r_1 t_{1,j} + r_2 t_{2,j} < 1, \ j = 1,2. \quad (2)$$

Below we shall see that this assumption guarantees the ergodicity of the system. By (2), the case $r_1 = r_2 = 1$ must be excluded.

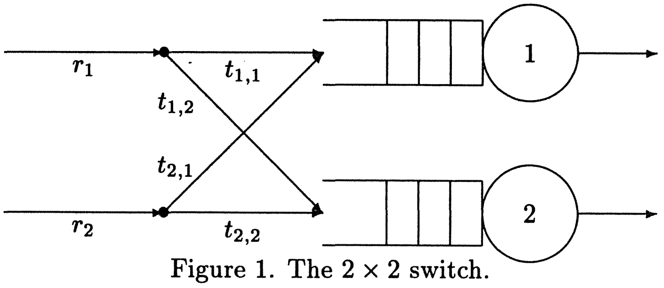


Figure 1. The 2 × 2 switch.

The behaviour of the 2×2 switch is described by a Markov chain with states (m, n), where m and n denote the numbers of waiting jobs at server 1 and server 2 at the beginning of a time unit. For a state (m, n) in the interior of the state space, we only have transitions to the neighbouring states $(m+m', n+n')$ with $m', n' \in \{-1, 0, 1\}$ and $m' + n' \leq 0$. The corresponding transition probabilities $q_{m',n'}$ are equal to:

$$\begin{aligned}
q_{1,-1} &= r_1 r_2 t_{1,1} t_{2,1}, \\
q_{0,0} &= r_1 r_2 (t_{1,1} t_{2,2} + t_{1,2} t_{2,1}), \\
q_{-1,1} &= r_1 r_2 t_{1,2} t_{2,2}, \\
q_{0,-1} &= r_1(1 - r_2) t_{1,1} + r_2(1 - r_1) t_{2,1}, \\
q_{-1,0} &= r_1(1 - r_2) t_{1,2} + r_2(1 - r_1) t_{2,2}, \\
q_{-1,-1} &= (1 - r_1)(1 - r_2).
\end{aligned}$$

Each transition probability for the states at the boundaries can be written as a sum of the probabilities $q_{m',n'}$. In figure 2 all transition probabilities, except the ones for the transitions from a state to itself, are illustrated.

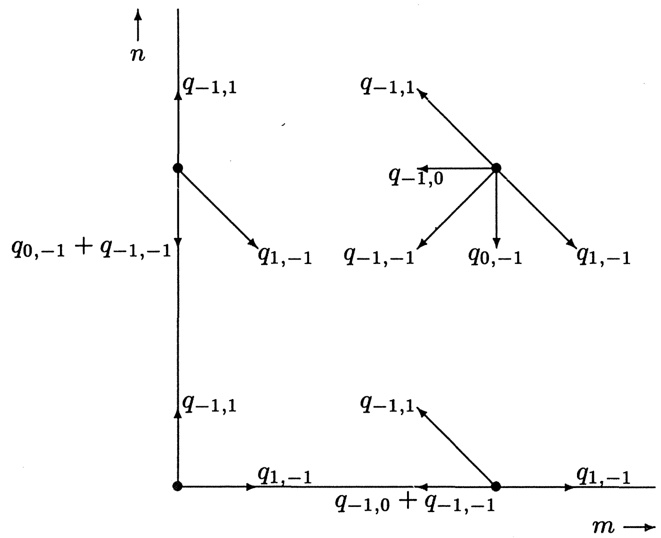


Figure 2. The transition probabilities.

Because all probabilities $q_{m',n'}$ except $q_{-1,-1}$ are guaranteed to be positive, the Markov chain and also the component (projected) chains, which describe the numbers of jobs present at one particular server, are irreducible and aperiodic. Due to assumption (2), which is equivalent with

$$q_{1,-1} < q_{-1,1} + q_{-1,0} + q_{-1,-1}, \tag{3}$$

for $j = 1$ and with

$$q_{-1,1} < q_{1,-1} + q_{0,-1} + q_{-1,-1}$$

for $j = 2$, the component Markov chains are random walks on $\mathbb{Z}_+$ with a negative drift and thus they are ergodic. This implies that the full Markov chain is also ergodic (see also Malyshev [14] for the ergodicity conditions of two-dimensional random walks with bounded jumps).

Let $\{p_{m,n}\}$, $\{p_k^{(1)}\}$ and $\{p_k^{(2)}\}$ be the equilibrium distributions of the full Markov chain and the two component chains. The distributions $\{p_k^{(1)}\}$ and $\{p_k^{(2)}\}$ are the marginal distributions of $\{p_{m,n}\}$:

$$p_k^{(1)} = \sum_{n=0}^{\infty} p_{k,n}, \; k \geq 0, \tag{4}$$

$$p_k^{(2)} = \sum_{m=0}^{\infty} p_{m,k}, \; k \geq 0.$$

Explicit formulae for the marginal distributions are easily derived, for example from figure 2 by using the balance principle

$$\text{the rate out of a set } V = \text{the rate into the set } V \tag{5}$$

for the sets $\{(m,n)|m \geq 0, n \geq k\}$, $k \geq 1$, and the sets $\{(m,n)|m \geq k, n \geq 0\}$, $k \geq 1$. We get the following geometric distributions:

$$p_k^{(1)} = \left(1 - \frac{q_{1,-1}}{q_{-1,1} + q_{-1,0} + q_{-1,-1}}\right)\left(\frac{q_{1,-1}}{q_{-1,1} + q_{-1,0} + q_{-1,-1}}\right)^k, \; k \geq 0, \tag{6}$$

$$p_k^{(2)} = \left(1 - \frac{q_{-1,1}}{q_{1,-1} + q_{0,-1} + q_{-1,-1}}\right)\left(\frac{q_{-1,1}}{q_{1,-1} + q_{0,-1} + q_{-1,-1}}\right)^k, \; k \geq 0. \tag{7}$$

For the distribution $\{p_{m,n}\}$ we have the following characterization: $\{p_{m,n}\}$ is the unique normalized solution of the equilibrium equations

$$q p_{m,n} = q_{1,-1} p_{m-1,n+1} + q_{-1,1} p_{m+1,n-1} + q_{0,-1} p_{m,n+1}$$

$$+ q_{-1,0} p_{m+1,n} + q_{-1,-1} p_{m+1,n+1} \text{ if } m > 0, n > 0, \tag{8}$$

$$(q - q_{0,-1}) p_{m,0} = q_{1,-1} p_{m-1,1} + q_{1,-1} p_{m-1,0} + q_{0,-1} p_{m,1}$$

$$+ (q_{-1,0} + q_{-1,-1}) p_{m+1,0} + q_{-1,-1} p_{m+1,1} \text{ if } m > 0, n = 0, \tag{9}$$

$$(q - q_{-1,0}) p_{0,n} = q_{-1,1} p_{1,n-1} + q_{-1,1} p_{0,n-1} + q_{-1,0} p_{1,n}$$

$$+ (q_{0,-1} + q_{-1,-1}) p_{0,n+1} + q_{-1,-1} p_{1,n+1} \text{ if } m = 0, n > 0, \tag{10}$$

$$(q_{1,-1}+q_{-1,1})p_{0,0}=(q_{-1,0}+q_{-1,-1})p_{1,0}+(q_{0,-1}+q_{-1,-1})p_{0,1}$$

$$+q_{-1,-1}p_{1,1} \text{ if } m=0, n=0, \tag{11}$$

where

$$q:=q_{1,-1}+q_{-1,1}+q_{0,-1}+q_{-1,0}+q_{-1,-1}. \tag{12}$$

The objective of the analysis in the next two sections is to derive explicit formulae for the equilibrium distribution $\{p_{m,n}\}$. Out of this equilibrium distribution one could easily compute interesting quantities such as the distribution and the moments of the total number of jobs in the system and the coefficient of correlation of the two queue lengths.

Remark 1. The particularly simple form of the marginal distributions in (6) and (7) is due to the 'projection property' (property (iv) in section 1). For example, note that in figure 2 for all $m \geq 0$ the total flow rate to the right equals $q_{1,-1}$ and for all $m > 0$ the total flow rate to the left equals $q_{-1,1}+q_{-1,0}+q_{-1,-1}$.
Remark 2. In the sequel we consider the class of random walks with transition probabilities as depicted in figure 2 (hence with the projection property). It should be noted that the random walk for the 2 × 2 switch is only a special element of this class. For example, one cannot realize $q_{1,-1}=q_{0,0}=q_{-1,1}$ with any choice of the r_i and $t_{i,j}$, but the analysis to be presented in sections 3 and 4 does hold for such a case. In fact, for the analysis below it is only required that $0<q_{1,-1}<q_{-1,1}+q_{-1,0}+q_{-1,-1}$ and $0<q_{-1,1}<q_{1,-1}+q_{0,-1}+q_{-1,-1}$.

3 THE COMPENSATION APPROACH

As already announced in the introduction, we shall use the *compensation approach* of Adan, Wessels and Zijm [6] to solve our problem. By using a compensation approach, Adan et al. try to determine the equilibrium distribution of a Markov process or Markov chain with state space $\mathbb{Z}_+^2$, transitions to neighbouring states only and uniform transition rates for all states in the interior of the state space, for all states at the horizontal boundary and for all states at the vertical boundary. They show that the application of their compensation approach is successful if and only if the Markov process or Markov chain is ergodic and if for all states in the interior no transitions are possible to the North, the North-East and the East. It is obvious that our Markov chain has the above structure and satisfies both conditions for a successful application.

When applying the compensation approach to our problem, we get several simplifications in the analysis due to the fact that the projection property holds (property (iv) in section 1). Roughly stated, for our Markov chain the set of transitions at the horizontal boundary as well as at the vertical boundary is a kind of projection of the set of transitions in the interior (see figure 2). In the same way, the set of transitions at the origin is the projection of both the transitions at the horizontal boundary and the transitions at the vertical boundary. Remark that the simple formulae (6) and (7) for the marginal distributions $\{p_k^{(1)}\}$ and $\{p_k^{(2)}\}$ were already a consequence of this projection property.

Because of the resulting simplifications, it is useful to briefly repeat the whole analysis of Adan et al. [6], but specified for our problem. Only for a small technical part we shall refer to [6]. In short, the latter analysis goes as follows. In this section we construct two series of product forms, which are *formal solutions* of the equilibrium equations. Here the adjective formal is used since we do not pay attention to the convergence of these series during the construction process. The convergence of the formal solutions is discussed in section 4. In that section we also show that the equilibrium distribution $\{p_{m,n}\}$ is found by simply taking the sum of the two formal solutions.

A formal solution is a linear combination of nonnull product forms $\alpha^m\beta^n$. It consists of an *initial term*, which satisfies the equilibrium equations in the interior of the state space and at one of the two boundaries, and a denumerable number of *compensation terms*. Each compensation term corrects the error made by the previous term at one of the two boundaries. Because later in the analysis the formal solutions have to be normalized to get the equilibrium distribution, the product forms $\alpha^m\beta^n$ are required to have factors α and β which lie inside the unit disk of the set of complex numbers.

The number of formal solutions needed for the equilibrium distribution equals the number of different initial terms that can be found. For the general case discussed in [6] one finds at least one and at most four initial terms. For our problem there exist two initial terms; one for the horizontal boundary and one for the vertical boundary. Contrary to the general case, we have explicit formulae for the initial terms.

Lemma 1.

(i) There exists exactly one product form $\alpha^m\beta^n$, $0 < |\alpha| < 1$ and $0 < |\beta| < 1$, which satisfies the equilibrium equations (8) and (9). The factors α and β of this product form are equal to

$$\alpha = \frac{q_{1,-1}}{q_{-1,1} + q_{-1,0} + q_{-1,-1}}, \tag{13}$$

$$\beta = \frac{q_{-1,1}\alpha^2}{q_{1,-1} + q_{0,-1}\alpha + q_{-1,-1}\alpha^2}. \tag{14}$$

(ii) There exists exactly one product form $\alpha^m\beta^n$, $0 < |\alpha| < 1$ and $0 < |\beta| < 1$, which satisfies the equilibrium equations (8) and (10). The factors β and α of this product form are equal to

$$\beta = \frac{q_{-1,1}}{q_{1,-1} + q_{0,-1} + q_{-1,-1}}, \tag{15}$$

$$\alpha = \frac{q_{1,-1}\beta^2}{q_{-1,1} + q_{-1,0}\beta + q_{-1,-1}\beta^2}. \tag{16}$$

Proof.

We only prove part (i). Part (ii) can be proved along the same lines. Let $\alpha^m\beta^n$, $0 < |\alpha| < 1$ and $0 < |\beta| < 1$, be a solution of (8) and (9). Substitution of $\alpha^m\beta^n$ in (8) and (9) gives

$$q\alpha\beta = q_{1,-1}\beta^2 + q_{-1,1}\alpha^2 + q_{0,-1}\alpha\beta^2 + q_{-1,0}\alpha^2\beta + q_{-1,-1}\alpha^2\beta^2, \tag{17}$$

$$(q-q_{0,-1})\alpha = q_{1,-1}\beta + q_{1,-1} + q_{0,-1}\alpha\beta + (q_{-1,0}+q_{-1,-1})\alpha^2 + q_{-1,-1}\alpha^2\beta. \tag{18}$$

Multiplying both sides of (18) by β and subtracting from both sides of (17) leads to

$$q_{0,-1}\alpha\beta = q_{-1,1}\alpha^2 - q_{1,-1}\beta - q_{-1,-1}\alpha^2\beta,$$

which shows that β has to be taken as presented by (14). To find α, we first rearrange the terms of (17):

$$(q_{1,-1} + q_{0,-1}\alpha + q_{-1,-1}\alpha^2)\beta^2 + (q_{-1,0}\alpha^2 - q\alpha)\beta + q_{-1,1}\alpha^2 = 0.$$

Now, dividing by β and substituting (14) gives a quadratic equation for α:

$$q_{-1,1}\alpha^2 + (q_{-1,0}\alpha^2 - q\alpha) + (q_{1,-1} + q_{0,-1}\alpha + q_{-1,-1}\alpha^2) = 0.$$

Finally, rearranging terms and using (12) leads to

$$(q_{-1,1} + q_{-1,0} + q_{-1,-1})\alpha^2 - (q_{1,-1} + q_{-1,1} + q_{-1,0} + q_{-1,-1})\alpha + q_{1,-1} = 0.$$

This quadratic equation has two real solutions, namely $\alpha = 1$ and α as given by (13). Here $\alpha = 1$ is not feasible, but, by (3), the other solution is. By (4), also the related solution (14) for β is feasible. This completes the proof. □

As we will see in section 4, for large m the product form $\alpha^m\beta^n$ with α and β as given by (13) and (14) will be the dominating term (largest α-factor) of the equilibrium distribution $\{p_{m,n}\}$. So this product form describes the behaviour of $\{p_{m,n}\}$ for large m, which explains that the α of (13) equals the parameter of the marginal distribution $\{p_k^{(1)}\}$. In the same way it is explained that the β of (15) equals the parameter of the marginal distribution $\{p_k^{(2)}\}$.

For both initial terms we get a formal solution. Let us first consider the formal solution $\{x_{m,n}\}$ with initial term $c_0\alpha_0^m\beta_0^n$, where α_0 and β_0 are defined by (13) and (14) and c_0 is a nonnull constant. Instead of only giving explicit formulae for $\{x_{m,n}\}$, we prefer to start with showing the main ideas of the compensation approach by describing the construction process which leads to $\{x_{m,n}\}$. This also enables us to make clear why we get such simple formal solutions for our problem.

The initial term $c_0\alpha_0^m\beta_0^n$ satisfies the equilibrium equations (8) and (9) for the interior of the state space and the horizontal boundary. However, it violates equation (10) for the vertical boundary. Because we need a solution which satisfies all equilibrium equations, we add a compensation term $c_1\alpha^m\beta^n$ to the initial term, such that the sum of these two terms satisfies the equations (8) and (10). Since this compensation term generates a new error at the horizontal boundary, after this more compensation terms have to be added. To show the details of the construction of a compensation term, we give an extensive description of the first compensation step in the next paragraph. All other compensation terms are constructed in the same way.

In the first compensation step, our mission is to define c_1, α and β such that the linear combination $c_0\alpha_0^m\beta_0^n + c_1\alpha^m\beta^n$ satisfies the equilibrium equations (8) and (10). Substitution of the linear combination in (10) gives the condition

$$c_0K(\alpha_0,\beta_0)\beta_0^{n-1} + c_1K(\alpha,\beta)\beta^{n-1} = 0 \text{ for all } n \geq 1, \tag{19}$$

where

$$\begin{aligned} K(\alpha,\beta) = {} & q_{-1,1} - (q - q_{-1,0})\beta + (q_{0,-1} + q_{-1,-1})\beta^2 \\ & +(q_{-1,1} + q_{-1,0}\beta + q_{-1,-1}\beta^2)\alpha. \end{aligned} \tag{20}$$

Because $c_0\alpha_0^m\beta_0^n$ violates (10), $K(\alpha_0,\beta_0) \neq 0$ and condition (19) forces us to take $\beta = \beta_0$. Next, we use equilibrium equation (8) for the choice for α. Because of the linearity of (8), $c_0\alpha_0^m\beta_0^n + c_1\alpha^m\beta_0^n$ is a solution of this equation if and only if $\alpha^m\beta_0^n$ is a solution. Substitution of $\alpha^m\beta_0^n$ in (8) (see (17)) and rearrangement of the terms gives the following quadratic equation for α:

$$(q_{-1,1} + q_{-1,0}\beta_0 + q_{-1,-1}\beta_0^2)\alpha^2 - (q\beta_0 - q_{0,-1}\beta_0^2)\alpha + q_{1,-1}\beta_0^2 = 0. \tag{21}$$

Of course, α_0 is one root of this quadratic equation. Let α_1 be the other root. Since we would have no compensation if we would take $\alpha = \alpha_0$, we have to take $\alpha = \alpha_1$. For the computation of α_1, one can use one of the formulae

$$\alpha_0\alpha_1 = \frac{q_{1,-1}\beta_0^2}{q_{-1,1} + q_{-1,0}\beta_0 + q_{-1,-1}\beta_0^2}, \tag{22}$$

$$\alpha_0 + \alpha_1 = \frac{q\beta_0 - q_{0,-1}\beta_0^2}{q_{-1,1} + q_{-1,0}\beta_0 + q_{-1,-1}\beta_0^2}, \tag{23}$$

which are the formulae for the product and the sum of the roots of the quadratic equation (21). Finally, we have to define the factor c_1. For all c_1 the linear combination $c_0\alpha_0^m\beta_0^n + c_1\alpha_1^m\beta_0^n$ satisfies (8), so our mission is completed if we define c_1 such that this linear combination satisfies (10). By (19), we find the expression

$$c_1 = -\frac{K(\alpha_0,\beta_0)}{K(\alpha_1,\beta_0)}c_0,$$

which can be simplified considerably due to the projection property. By substituting (23) in formula (20) for $K(\alpha_0,\beta_0)$ and $K(\alpha_1,\beta_0)$, we find due to the projection property

$$K(\alpha_0,\beta_0) = (1-\alpha_1)(q_{-1,1} + q_{-1,0}\beta_0 + q_{-1,-1}\beta_0^2),$$

$$K(\alpha_1,\beta_0) = (1-\alpha_0)(q_{-1,1} + q_{-1,0}\beta_0 + q_{-1,-1}\beta_0^2),$$

by which the expression for c_1 simplifies to

$$c_1 = -\frac{1-\alpha_1}{1-\alpha_0}c_0. \tag{24}$$

The solution $c_0\alpha_0^m\beta_0^n + c_1\alpha_1^m\beta_0^n$, which we have after the first compensation step, satisfies the equilibrium equations (8) and (10) for the interior and the vertical boundary. However, the compensation term $c_1\alpha_1^m\beta_0^n$ has generated a new error (a smaller one, as will be shown in section 4) at the horizontal boundary. To compensate for this error, we again have to add a compensation term, and so on. As a result, we get the following formal solution, where d_0 is a second nonnull constant:

$$x_{m,n} = \underbrace{\overbrace{c_0d_0\alpha_0^m\beta_0^n}^{H} + \rlap{$\overbrace{}^{H}$}c_1d_0\alpha_1^m\beta_0^n}_{V} + \underbrace{c_1d_1\alpha_1^m\beta_1^n + c_2d_1\alpha_2^m\beta_1^n}_{V} + \ldots$$

The construction is such that each term in this series satisfies (8), each sum of terms with the same α-factor satisfies (9) and each sum of terms with the same β-factor satisfies (10). As a consequence, $\{x_{m,n}\}$ satisfies the equilibrium equations (8), (9) and (10).

The formulae for $\{x_{m,n}\}$ are as follows. By taking pairs of product forms in two different ways, we get two expressions:

$$x_{m,n} = c_0d_0\alpha_0^m\beta_0^n + \sum_{i=0}^{\infty} c_{i+1}\alpha_{i+1}^m(d_i\beta_i^n + d_{i+1}\beta_{i+1}^n) \tag{25}$$

$$= \sum_{i=0}^{\infty} d_i\beta_i^n(c_i\alpha_i^m + c_{i+1}\alpha_{i+1}^m),\ m \geq 0, n \geq 0. \tag{26}$$

The factors α_0 and β_0 are given by (13) and (14). The other α- and β-factors are found by the quadratic equation (17) which is obtained by substituting the product form $\alpha^m\beta^n$ in the equilibrium equation (8) for the interior (compare (21) and (22)):

$$\alpha_{i+1} = \frac{q_{1,-1}\beta_i^2}{q_{-1,1} + q_{-1,0}\beta_i + q_{-1,-1}\beta_i^2} \cdot \frac{1}{\alpha_i},\ i \geq 0, \tag{27}$$

$$\beta_{i+1} = \frac{q_{-1,1}\alpha_{i+1}^2}{q_{1,-1} + q_{0,-1}\alpha_{i+1} + q_{-1,-1}\alpha_{i+1}^2} \cdot \frac{1}{\beta_i},\ i \geq 0. \tag{28}$$

The coefficients c_0 and d_0 are only required to be nonnull constants. Formulae for the other coefficients are derived in the same way as formula (24) for c_1:

$$c_{i+1} = -\frac{1-\alpha_{i+1}}{1-\alpha_i}c_i,\ i \geq 0, \tag{29}$$

$$d_{i+1} = -\frac{1-\beta_{i+1}}{1-\beta_i}d_i,\ i \geq 0. \tag{30}$$

The simple form of the recursive formulae (29) and (30) for the coefficients c_i and d_i (which is due to the projection property, cf. the derivation of formula (24) for c_1), leads to an elegant expression for $\{x_{m,n}\}$. Define $c_0 := 1 - \alpha_0$ and

$d_0 := 1 - \beta_0$, then the recursive formulae (29) and (30) for c_i and d_i are easily rewritten to

$$c_i = (-1)^i(1-\alpha_i),\ i \geq 0,$$
$$d_i = (-1)^i(1-\beta_i),\ i \geq 0.$$

Substitution of these formulae in (25) and (26) yields

$$x_{m,n} = (1-\alpha_0)\alpha_0^m(1-\beta_0)\beta_0^n - \sum_{i=0}^{\infty}(1-\alpha_{i+1})\alpha_{i+1}^m[(1-\beta_i)\beta_i^n - (1-\beta_{i+1})\beta_{i+1}^n] \tag{31}$$

$$= \sum_{i=0}^{\infty}(1-\beta_i)\beta_i^n[(1-\alpha_i)\alpha_i^m - (1-\alpha_{i+1})\alpha_{i+1}^m],\ m \geq 0, n \geq 0, \tag{32}$$

which show that $\{x_{m,n}\}$ is an alternating sum of two-dimensional product-form (geometric) probability distributions.

For the other formal solution $\{\tilde{x}_{m,n}\}$ generated by the initial term with factors defined by (15) and (16), we get similar expressions as for $\{x_{m,n}\}$:

$$\tilde{x}_{m,n} = (1-\tilde{\alpha}_0)\tilde{\alpha}_0^m(1-\tilde{\beta}_0)\tilde{\beta}_0^n - \sum_{i=0}^{\infty}(1-\tilde{\beta}_{i+1})\tilde{\beta}_{i+1}^n[(1-\tilde{\alpha}_i)\tilde{\alpha}_i^m - (1-\tilde{\alpha}_{i+1})\tilde{\alpha}_{i+1}^m] \tag{33}$$

$$= \sum_{i=0}^{\infty}(1-\tilde{\alpha}_i)\tilde{\alpha}_i^m[(1-\tilde{\beta}_i)\tilde{\beta}_i^n - (1-\tilde{\beta}_{i+1})\tilde{\beta}_{i+1}^n],\ m \geq 0, n \geq 0. \tag{34}$$

Here the factors $\tilde{\beta}_0$ and $\tilde{\alpha}_0$ are defined by (15) and (16). The other factors are defined such that each product form satisfies quadratic equation (17):

$$\tilde{\beta}_{i+1} = \frac{q_{-1,1}\tilde{\alpha}_i^2}{q_{1,-1} + q_{0,-1}\tilde{\alpha}_i + q_{-1,-1}\tilde{\alpha}_i^2} \cdot \frac{1}{\tilde{\beta}_i},\ i \geq 0, \tag{35}$$

$$\tilde{\alpha}_{i+1} = \frac{q_{1,-1}\tilde{\beta}_{i+1}^2}{q_{-1,1} + q_{-1,0}\tilde{\beta}_{i+1} + q_{-1,-1}\tilde{\beta}_{i+1}^2} \cdot \frac{1}{\tilde{\alpha}_i},\ i \geq 0. \tag{36}$$

This completes the description of the two formal solutions $\{x_{m,n}\}$ and $\{\tilde{x}_{m,n}\}$, which, formally, both are solutions of the equilibrium equations (8), (9) and (10).

4 THE MAIN THEOREM

In this section we prove the main result of this paper, which states that the equilibrium distribution $\{p_{m,n}\}$ is equal to the sum of the two formal solutions. The proof consists of two parts. First, we prove that the formal solutions converge in all states except in the origin. For this, we shall refer to some results of Adan et al. [6]. In the second part, it is shown that the sum of the two formal solutions satisfies all equilibrium equations. For this we could also use a result of Adan et al. [6], however, we prefer to give an alternative proof.

For the convergence of the formal solutions $\{x_{m,n}\}$ and $\{\tilde{x}_{m,n}\}$, we need information about the limiting behaviour of the α- and β-factors. For this we gather some results of the analysis in [6].

Lemma 2. *For the factors α_i, β_i, $\tilde{\alpha}_i$ and $\tilde{\beta}_i$ we have:*

(i) $1 > \alpha_0 > \beta_0 > \alpha_1 > \beta_1 > \ldots \downarrow 0$;

(ii) $\frac{\alpha_{i+1}}{\beta_i} \to A_1$ *and* $\frac{\beta_i}{\alpha_i} \to \frac{1}{A_2}$ *as* $i \to \infty$; (37)

(iii) $1 > \tilde{\beta}_0 > \tilde{\alpha}_0 > \tilde{\beta}_1 > \tilde{\alpha}_1 > \ldots \downarrow 0$;

(iv) $\frac{\tilde{\beta}_{i+1}}{\tilde{\alpha}_i} \to \frac{1}{A_2}$ *and* $\frac{\tilde{\alpha}_i}{\tilde{\beta}_i} \to A_1$ *as* $i \to \infty$; (38)

Here, A_1 *and* A_2 *are defined by*

$$A_1 = \frac{q - \sqrt{q^2 - 4q_{1,-1}q_{-1,1}}}{2q_{-1,1}}, \quad A_2 = \frac{q + \sqrt{q^2 - 4q_{1,-1}q_{-1,1}}}{2q_{-1,1}}. \tag{39}$$

Part (i) of this lemma is proved by first writing the quadratic equation (17), by which the α- and β-factors are defined, as an equation in $z = \beta/\alpha$ and then applying Rouché's theorem (see lemma 4.3 of [6]). Part (ii) is found by first writing the roots of the above quadratic equation in $z = \beta/\alpha$ as function of α and then letting $\alpha \to 0$ (see lemma 6.1 of [6]). The parts (iii) and (iv) are proved along the same lines.

By lemma 2,

$$\frac{(1-\alpha_{i+1})\alpha_{i+1}^m(1-\beta_{i+1})\beta_{i+1}^n}{(1-\alpha_i)\alpha_i^m(1-\beta_i)\beta_i^n} \to \left(\frac{A_1}{A_2}\right)^{m+n} \tag{40}$$

and

$$\frac{(1-\alpha_{i+2})\alpha_{i+2}^m(1-\beta_{i+1})\beta_{i+1}^n}{(1-\alpha_{i+1})\alpha_{i+1}^m(1-\beta_i)\beta_i^n} \to \left(\frac{A_1}{A_2}\right)^{m+n} \tag{41}$$

as $i \to \infty$. Elementary algebra shows that A_1 and A_2 are positive reals with $A_1 < A_2$. Therefore the limits in (40) and (41) are smaller than 1 for all $m \geq 0$, $n \geq 0$ and $m + n \geq 1$, which proves part (i) of the following lemma. Part (ii) of that lemma, which is needed in the second part of this section, is also easily proved by using lemma 2.

Lemma 3. *For* $\{x_{m,n}\}$ *we have:*

(i) The series

$$\sum_{i=0}^{\infty}(1-\alpha_i)\alpha_i^m(1-\beta_i)\beta_i^n and \sum_{i=0}^{\infty}(1-\alpha_{i+1})\alpha_{i+1}^m(1-\beta_i)\beta_i^n$$

are absolutely convergent for all $m \geq 0$, $n \geq 0$ *and* $m+n \geq 1$. *So,* $\{x_{m,n}\}$ *is well defined by (31) and (32) in all states except in the origin.*

(ii)

$$\sum_{\substack{m \geq 0, n \geq 0 \\ m+n \geq 1}} |x_{m,n}| < \infty. \tag{42}$$

The same results hold for $\{\tilde{x}_{m,n}\}$.

By lemma 3, $\{x_{m,n}\}$ and $\{\tilde{x}_{m,n}\}$ satisfy the equilibrium equations in all states except in $(0,0)$, $(1,0)$ and $(0,1)$.

From the analysis in Adan et al. [6], we know that the equilibrium distribution is found by taking a linear combination of the formal solutions, of which the coefficients can be determined by substituting this linear combination in some equilibrium equations. This is proved by analyzing the embedded process on the set of states where the formal solutions are absolutely convergent. For our problem, however, due to the projection property, it is clear that we have to take the sum of the formal solutions and we can give an alternative proof to show that the equilibrium distribution is equal to this sum.

Due to the projection property, we found the formulae (31) till (34). Using these formulae, we easily see

$$\sum_{n=0}^{\infty} x_{m,n} = (1-\alpha_0)\alpha_0^m \text{ for all } m \geq 1, \ \sum_{m=0}^{\infty} x_{m,n} = 0 \text{ for all } n \geq 1,$$

$$\sum_{n=0}^{\infty} \tilde{x}_{m,n} = 0 \text{ for all } m \geq 1, \ \sum_{m=0}^{\infty} \tilde{x}_{m,n} = (1-\tilde{\beta}_0)\tilde{\beta}_0^n \text{ for all } n \geq 1.$$

Since α_0 and $\tilde{\beta}_0$ are equal to the parameters of the marginal distributions $\{p_k^{(1)}\}$ and $\{p_k^{(2)}\}$ (see also the paragraph right after the proof of lemma 1), by defining $\{\tilde{p}_{m,n}\}$ as the sum of the formal solutions, i.e.

$$\tilde{p}_{m,n} := x_{m,n} + \tilde{x}_{m,n}, \ m \geq 0, n \geq 0, m+n \geq 1, \tag{43}$$

we obtain a solution for which

$$\sum_{n=0}^{\infty} \tilde{p}_{m,n} = p_m^{(1)} \text{ for all } m \geq 1, \ \sum_{m=0}^{\infty} \tilde{p}_{m,n} = p_n^{(2)} \text{ for all } n \geq 1. \tag{44}$$

To get a distribution, we define $\tilde{p}_{0,0}$ by

$$\tilde{p}_{0,0} := 1 - \sum_{\substack{m\geq 0, n\geq 0 \\ m+n\geq 1}} \tilde{p}_{m,n}, \tag{45}$$

which is a correct definition due to lemma 3. Now, rewriting $\tilde{p}_{0,0}$ leads to

$$\tilde{p}_{0,0} = (1-\alpha_0) - \sum_{n=1}^{\infty} \tilde{p}_{0,n} = (1-\tilde{\beta}_0) - \sum_{m=1}^{\infty} \tilde{p}_{m,0}.$$

Hence the first equality in (44) also holds for $m = 0$ and the second equality in (44) also holds for $n = 0$. As a consequence, the marginal distributions of $\{\tilde{p}_{m,n}\}$ are equal to the marginal distributions $\{p_k^{(1)}\}$ and $\{p_k^{(2)}\}$ of $\{p_{m,n}\}$:

$$\sum_{n=0}^{\infty} \tilde{p}_{m,n} = p_m^{(1)} \text{ for all } m \geq 0, \ \sum_{m=0}^{\infty} \tilde{p}_{m,n} = p_n^{(2)} \text{ for all } n \geq 0. \tag{46}$$

Since $\{\tilde{p}_{m,n}\}$ is a linear combination of the formal solutions, we know $\{\tilde{p}_{m,n}\}$ satisfies the equilibrium equations in all states except in (0,0), (1,0) and (0,1). To show that $\{\tilde{p}_{m,n}\}$ also satisfies the equations in these remaining states we use (46) and the balance principle (5). The balance principle for the set $V_1 = \{(m,n)|m \geq 1, n \geq 0\}$ gives the condition

$$(q_{-1,1} + q_{-1,0} + q_{-1,-1}) \sum_{n=0}^{\infty} p_{1,n} = q_{1,-1} \sum_{n=0}^{\infty} p_{0,n}.$$

By using (46) it is easily shown that $\{\tilde{p}_{m,n}\}$ satisfies this condition, i.e. the balance principle for the set V_1. However, $\{\tilde{p}_{m,n}\}$ also satisfies the balance principle for the subset $V_2 = \{(m,n)|m \geq 1, n \geq 0\}\backslash\{(1,0)\}$ of V_1, since it satisfies the balance principle (i.e. the equilibrium equation) for every state of this set. But then $\{\tilde{p}_{m,n}\}$ also satisfies the balance principle for $V_1\backslash V_2$, i.e. the equilibrium equation in (1,0). In the same way it is proved that $\{\tilde{p}_{m,n}\}$ satisfies the equations in (0,1) and (0,0). So, $\{\tilde{p}_{m,n}\}$ satisfies all equilibrium equations. This completes the proof of our main theorem, which states that the equilibrium joint queue length distribution $\{p_{m,n}\}$ can be written as the sum of two alternating series of two-dimensional product-form geometric distributions.

Main theorem. *For all $m \geq 0$, $n \geq 0$ and $m + n \geq 1$:*

$$p_{m,n} = x_{m,n} + \tilde{x}_{m,n} \tag{47}$$

$$= \sum_{i=0}^{\infty}(1-\beta_i)\beta_i^n[(1-\alpha_i)\alpha_i^m - (1-\alpha_{i+1})\alpha_{i+1}^m]$$

$$+ \sum_{i=0}^{\infty}(1-\tilde{\alpha}_i)\tilde{\alpha}_i^m[(1-\tilde{\beta}_i)\tilde{\beta}_i^n - (1-\tilde{\beta}_{i+1})\tilde{\beta}_{i+1}^n]. \tag{48}$$

Due to the fact that the sequences $\{\alpha_{i+1}/\beta_i\}$, $\{\beta_i/\alpha_i\}$, $\{\tilde{\beta}_{i+1}/\tilde{\alpha}_i\}$ and $\{\tilde{\alpha}_i/\tilde{\beta}_i\}$, of which the limits are given in (37) and (38), are monotonously decreasing (see lemma 5.1 of [6]), one can easily derive tight error bounds for the two series in (48). Together with these bounds the result stated in the main theorem leads to a very efficient algorithm for the computation of $\{p_{m,n}\}$. Next to $\{p_{m,n}\}$, the main result stated in (48) also leads to efficient algorithms for interesting quantities such as the total number of jobs in the system and the coefficient of correlation of the two queue lengths.

5 THE COEFFICIENT OF CORRELATION OF THE TWO QUEUE LENGTHS

For the 2 × 2 switch it is interesting to learn something about the correlation of the two queue lengths. Let Q_i be the queue length at server i, $i = 1, 2$, and let $\rho(Q_1, Q_2)$ be the coefficient of correlation of Q_1 and Q_2:

$$\rho(Q_1, Q_2) = \frac{E\{Q_1 Q_2\} - EQ_1 EQ_2}{\sigma(Q_1)\sigma(Q_2)}. \tag{49}$$

Because Q_1 and Q_2 are geometric distributions with parameters α_0 and $\tilde{\beta}_0$ respectively, the expectation and standard deviation of Q_1 and Q_2 are given by

$$EQ_1 = \frac{\alpha_0}{1-\alpha_0}, \ \sigma(Q_1) = \frac{\sqrt{\alpha_0}}{1-\alpha_0},$$

$$EQ_2 = \frac{\tilde{\beta}_0}{1-\tilde{\beta}_0}, \ \sigma(Q_2) = \frac{\sqrt{\tilde{\beta}_0}}{1-\tilde{\beta}_0}.$$

A formula for the expectation of the product of the two queue lengths is easily found by using (48):

$$E\{Q_1Q_2\} = \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} mnp_{m,n}$$

$$= \sum_{i=0}^{\infty} \frac{\beta_i}{1-\beta_i} \frac{\alpha_i - \alpha_{i+1}}{(1-\alpha_i)(1-\alpha_{i+1})} + \sum_{i=0}^{\infty} \frac{\tilde{\alpha}_i}{1-\tilde{\alpha}_i} \frac{\tilde{\beta}_i - \tilde{\beta}_{i+1}}{(1-\tilde{\beta}_i)(1-\tilde{\beta}_{i+1})}. \tag{50}$$

We have used the formulae (49) and (50) to compute the coefficient of correlation for the symmetric 2 × 2 buffered switch. For this symmetric switch the input parameters are given by

$$r_1 = r_2 = p, \ 0 < p < 1, \tag{51}$$

$$t_{1,1} = t_{1,2} = t_{2,1} = t_{2,2} = 1/2. \tag{52}$$

In this case p is equal to the workload of the system and the coefficient of correlation is only a function of this p, i.e. $\rho(Q_1, Q_2) = \rho(p)$. The function $\rho(p)$ is pictured in figure 3. The results needed for this figure have been computed on an IBM-compatible PC\AT in less than one second.

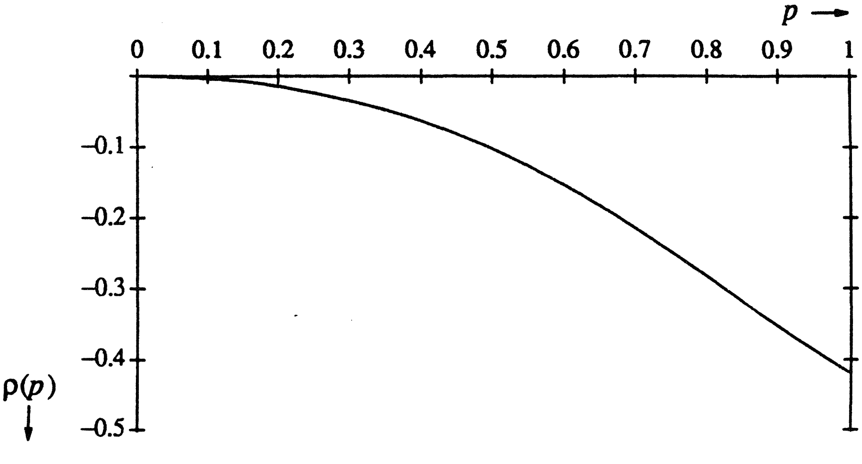


Figure 3. The coefficient of correlation of the two queue lengths for the symmetric 2×2 switch.

As we see in figure 3, we have a negative correlation for all p. Intuitively this negative correlation is not surprising, since there is a negative correlation $-p/(2-p)$ between the numbers of jobs which arrive at the two servers at the beginning of a certain time unit. Figure 3 now tells us that the negative correlation for the two queue lengths is very weak for low workloads p. It may be easily shown that $\rho(p) \approx -\frac{1}{4}p^2$ as $p \downarrow 0$. For higher workloads p the negative correlation gets rather strong. The maximal strength of the correlation is reached for p close to 1: $\rho(p) \to -0.4203$ as $p \uparrow 1$. Indeed, using the formulae (27), (28) for α_i, β_i one can show that for $p \uparrow 1$:

$$\alpha_i = 1 - 4(i+1)(2i+1)(1-p) + o(1-p),$$

$$\beta_i = 1 - 4(i+1)(2i+3)(1-p) + o(1-p).$$

Hence from (49) and (50) (note that in the symmetric case $\tilde{\beta}_i = \alpha_i$ and $\tilde{\alpha}_i = \beta_i$ for all i):

$$\lim_{p\uparrow 1} \rho(p) = -1 + 2\sum_{i=0}^{\infty} \frac{1}{(i+1)(2i+3)}[\frac{1}{(i+1)(2i+1)} - \frac{1}{(i+2)(2i+3)}].$$

Rewriting,

$$\lim_{p\uparrow 1} \rho(p) = -1 + 2\sum_{k=1}^{\infty} \frac{1}{k^2(2k-1)(2k+1)} - 2\sum_{k=1}^{\infty} \frac{1}{k(k+1)(2k+1)^2}$$

$$= -1 + 8\sum_{k=1}^{\infty} \frac{1}{(2k-1)(2k+1)} - 2\sum_{k=1}^{\infty} \frac{1}{k^2} - 2\sum_{k=1}^{\infty} \frac{1}{k(k+1)} + 8\sum_{k=1}^{\infty} \frac{1}{(2k+1)^2}$$

$$= -1 + 4 - \frac{\pi^2}{3} - 2 + 8(\frac{\pi^2}{8} - 1)$$

$$= \frac{2}{3}\pi^2 - 7.$$

This leading term for the heavy traffic behaviour of the queue length correlation coefficient also follows from section 5 of Jaffe [12].

6 COMPLEX-VARIABLE METHODS

As observed in the introduction, Jaffe [11,12] has analyzed the symmetric 2 × 2 clocked buffered switch problem by two different complex-variable methods, viz., the boundary value method and the uniformization technique of Flatto and McKean. In the present section we briefly outline these two solutions, and we point out some differences and similarities with the compensation approach.

In the symmetric case the input parameters are given by (51) and (52). In both complex-variable methods the first step is the introduction of the generating function

$$f(x,y) := \sum_{m=0}^{\infty}\sum_{n=0}^{\infty} p_{m,n} x^m y^n, \; |x| \le 1, |y| \le 1. \tag{53}$$

It follows from (8)-(11) that, for $|x| \leq 1$, $|y| \leq 1$, $f(x,y)$ satisfies the following functional equation:

$$(xy - r(x,y))f(x,y) = (y-1)r(x,0)f(x,0) + (x-1)r(0,y)f(0,y) \\ +(x-1)(y-1)r(0,0)f(0,0), \qquad (54)$$

where

$$r(x,y) := (1 - p + \frac{p}{2}(x+y))^2. \qquad (55)$$

Denote by S the complex curve $xy - r(x,y) = 0$ (the zeroes of the 'kernel' of (54)), by D the interior of the unit circle, by $\bar{D}$ the closure of the unit circle; and subsequently separate the x- and y-parts of the righthand side of (54) by defining

$$g(x) := \frac{r(x,0)f(x,0)}{x-1} + \frac{1}{2}r(0,0)f(0,0) = \frac{r(0,x)f(0,x)}{x-1} + \frac{1}{2}r(0,0)f(0,0). \qquad (56)$$

Then the boundedness of $f(x,y)$ in $\bar{D}^2$ implies that

$$g(x) + g(y) = 0, \ (x,y) \in S \cap \bar{D}^2, \ x,y \neq 1, \qquad (57)$$

while, as seen from (56), g has a simple pole at 1. Formula (57) is the fundamental equation in both complex-variable methods, which are successively discussed below.

A. The boundary value method

We outline the approach in Jaffe [11]. The boundary value method considers a suitable subset of $S \cap \bar{D}^2$, by taking $y = \bar{x}$. Formula (57) now reduces to

$$g(x) + g(\bar{x}) = 0, \ x \in E \backslash \{1\}. \qquad (58)$$

Here E is the ellipse $\{x : |x|^2 = r(x,\bar{x}), x \in D\}$. Let ϕ, with inverse ψ, be the conformal mapping of the unit disk onto the region bounded by E, with normalization conditions $\phi(0) = p/(1+p)$, $\phi(1) = 1$. Define $h(w) := g(\phi(w))$. We now obtain a "boundary value problem with a pole" of an extremely simple form, cf. Section I.3.3 of [8], for $h(.)$ on the unit circle Γ:

$$\mathrm{Re}\, h(w) = 0, \ w \in \Gamma \backslash \{1\}, \qquad (59)$$

$$\lim_{w \to 1} (w-1)h(w) = \frac{1-p}{\phi'(1)},$$

with $h(.)$ analytic on D, continuous on $\bar{D}\backslash\{1\}$. The solution of this boundary value problem is

$$h(w) = \frac{1}{2}\frac{1-p}{\phi'(1)}\frac{w+1}{w-1}, \ w \in D, \qquad (60)$$

which determines $g(x) = h(\psi(x))$; here the conformal mapping $\psi(x)$ is explicitly expressed in the Jacobi elliptic (sin *am* or *sn*) function. Substitution of

$$g(u) = \frac{1}{2}\frac{1-p}{\phi'(1)}\frac{\psi(u)+1}{\psi(u)-1}$$

in (54) (for $u = x, y$) finally yields $f(x,y)$, for $|x| \leq 1$, $|y| \leq 1$:

$$f(x,y) = (1-p)\psi'(1)\frac{(x-1)(y-1)}{(\psi(x)-1)(\psi(y)-1)}\frac{\psi(x)\psi(y)-1}{xy-r(x,y)}.$$

B. The uniformization technique
Starting point is again formula (57). Jaffe [12] exploits the following idea. Suppose that g is meromorphic, i.e., all its singularities are isolated poles. It follows from (56) that g has a simple pole at 1, with residue $1-p$. For points $(x,y) \in S$, a simple pole of $g(x)$ at x with known residue must be compensated by a simple pole of $g(y)$ at y, with residue being determined by (57). Starting from the pole at 1 one now iteratively determines a countable set of poles. Jaffe shows that g has simple poles at

$$b_m := \tilde{\alpha}\lambda^m + \tilde{\beta}\lambda^{-m} + \tilde{\gamma}, \; m = 0, 1, ..., \tag{61}$$

with residues

$$a_m := (-1)^m\sqrt{1-p^2}(\tilde{\alpha}\lambda^m - \tilde{\beta}\lambda^{-m}); \tag{62}$$

here

$$\tilde{\alpha} := \frac{1+\sqrt{1-p^2}}{2(1+p)}, \; \tilde{\beta} := \frac{1-\sqrt{1-p^2}}{2(1+p)}, \; \tilde{\gamma} := \frac{p}{1+p}, \; \lambda := \frac{\tilde{\alpha}}{\tilde{\beta}}.$$

Observe that $b_0 = 1 < b_1 < b_2 < ....$ For later reference we note that

$$b_1 = (\frac{2}{p} - 1)^2, \; b_2 = (1 + \frac{2}{p} - \frac{4}{p^2})^2. \tag{63}$$

Jaffe derives (61) and (62) by first introducing linear coordinate changes for the hyperbola $xy - r(x,y) = 0$:

$$x = \tilde{\alpha}\tilde{x} + \tilde{\beta}\tilde{y} + \tilde{\gamma}, \; y = \tilde{\alpha}\tilde{y} + \tilde{\beta}\tilde{x} + \tilde{\gamma},$$

which transform S into $\tilde{x}\tilde{y} = 1$, or $\tilde{y} = 1/\tilde{x}$. Hence $\tilde{x}$ is a uniformizing variable which parameterizes S. Jaffe shows that the transformed version of (57) gives rise to simple poles at $\tilde{x} = \lambda^m$, $m = 0, 1, ...$, which implies that g has simple poles given by (61). He finally verifies that his initial assumption of g being a meromorphic function is indeed correct.

Comparison between the compensation approach and the uniformization technique
Using the compensation approach we have $\tilde{\beta}_i = \alpha_i$ and $\tilde{\alpha}_i = \beta_i$ for the completely symmetric case. Hence from (48) the generating function of $\{p_{m,n}\}$ is given by

$$p_{0,0} + \sum_{i=0}^{\infty}\frac{1-\beta_i}{1-\beta_i y}[\frac{(1-\alpha_i)\alpha_i x}{1-\alpha_i x} - \frac{(1-\alpha_{i+1})\alpha_{i+1}x}{1-\alpha_{i+1}x} - (\alpha_i - \alpha_{i+1})\beta_i y]$$

$$+\sum_{i=0}^{\infty}\frac{1-\beta_i}{1-\beta_i x}[\frac{(1-\alpha_i)\alpha_i y}{1-\alpha_i y} - \frac{(1-\alpha_{i+1})\alpha_{i+1}y}{1-\alpha_{i+1}y} - (\alpha_i - \alpha_{i+1})\beta_i x]. \tag{64}$$

Observe that this generating function, and the ones for $\{p_{m,0}\}$ and $\{p_{0,n}\}$, are meromorphic functions with simple poles at $1/\alpha_i$, $i = 0, 1, \ldots$ and at $1/\beta_i$, $i = 0, 1, \ldots$. We claim that the sequence $\{1/\alpha_0, 1/\beta_0, 1/\alpha_1, 1/\beta_1, \ldots\}$ corresponds to the sequence $\{b_1, b_2, b_3, b_4, \ldots\}$. From (13) and (14) it is seen that

$$\alpha_0 = (\frac{2}{p} - 1)^{-2}, \ \beta_0 = (1 + \frac{2}{p} - \frac{4}{p^2})^{-2}. \tag{65}$$

Comparison with (63) reveals that indeed $b_1 = 1/\alpha_0$ and $b_2 = 1/\beta_0$. In fact Jaffe [12] starts with $b_0 = 1$ and b_1, which corresponds to our observation, in the proof of Lemma 1, that the quadratic equation being there considered has two real solutions, viz., 1 and α_0. The successive α_i and β_i are determined from (27) and (28), what really amounts to finding product forms that satisfy the equilibrium equation (8) for the interior and that together with a previous product form satisfy one of the two equilibrium equations (9) and (10) for the boundaries. In terms of generating functions, this is translated into finding those zero tuples (x, y) of the 'kernel' $xy - r(x, y) = 0$ that are related via (57) (note that (i) the kernel $xy - r(x, y) = 0$ is completely determined by the behaviour of the random walk in the interior; (ii) the righthand side of (54) reflects the behaviour of the random walk on the boundaries; and (iii) demanding that (57) holds for points (x, y) that are zeroes of the kernel corresponds to demanding that the equilibrium equations are satisfied both in the interior and on the boundaries). Remember that in the compensation approach each time a new term is added, to compensate an error on one of the boundaries; in terms of generating functions, this is translated into adding a new pole b_m to compensate a pole b_{m-1} in (57). The above reasoning implies the following:

(i) The mechanism to find β_i for given α_i (or α_{i+1} for given β_i) is equivalent with Jaffe's mechanism to find b_m given b_{m-1}, viz., by solving the equation $b_{m-1}y - r(b_{m-1}, y) = 0$.

(ii) $b_{2m+1} = 1/\alpha_m, b_{2m+2} = 1/\beta_m$.

Hence we see that the generating function given by (64) has exactly the same (simple) poles as the generating function $f(x, y)$, and that in both approaches these poles, in increasing order of absolute value, are successively obtained from one another by compensating the effect of the preceding pole.

Comparison between the boundary value method and the uniformization technique

In the boundary value method, $g(x)$ has poles at the zeroes of $\psi(x) - 1 = 0$. The normalization condition $\phi(1) = 1$ for the conformal mapping implies $\psi(1) = 1$, so that $b_0 = 1$ is again found to be a pole of g. The periodic nature of the Jacobian elliptic function $\psi(.)$ subsequently leads to the sequence of poles $b_1, b_2, \ldots$.

From an analytic point of view, both complex-variable methods are for the present model of similar complexity (compared with the shortest queue problem and similar two-dimensional problems, one might say: of similar simplicity).

They lead to different representations of the two-dimensional queue-length generating function. From a numerical point of view these representations can be exploited to obtain, e.g., queue length moments; however, the explicit representation obtained by the compensation approach seems more suitable for numerical calculations.

In a forthcoming report Cohen [7] presents a fundamental discussion of complex-variable approaches to a class of two-dimensional random walks that contains that of the present paper. This is the class of positive recurrent semi-homogeneous nearest neighbouring (i.e., only one-step transition probabilities can be positive) random walks in the first quadrant, for which in the interior no transitions are allowed to the North, East and North-East (the class of random walks studied in [6]). He proves that the bivariate generating function of the stationary distribution of such two-dimensional random walks in the first quadrant can be represented by meromorphic functions. Subsequently he exposes the construction of those meromorphic functions; this construction is again based on the iterative calculation of poles and residues. Cohen [7] claims that the bivariate generating function for this class of random walks may also be obtained using the boundary value method, even when transitions to the North, East and North-East are allowed; but he remarks that when such transitions are excluded, then the construction of the meromorphic function via the iterative calculation of poles and residues is simpler because it avoids the explicit calculation of a conformal mapping.

Acknowledgment The authors are indebted to I.J.B.F. Adan, J.W. Cohen, and J. Wessels for fruitful discussions.

REFERENCES

1. Adan, I.J.B.F. (1991). A compensation approach for queueing problems. Ph.D. thesis, Eindhoven University of Technology, Eindhoven, The Netherlands.
2. Adan, I.J.B.F., van Houtum, G.J., Wessels, J., and Zijm, W.H.M. (1993). A compensation procedure for multiprogramming queues. OR Spektrum 15: 95-106.
3. Adan, I.J.B.F., Wessels, J., and Zijm, W.H.M. (1990). Analysis of the symmetric shortest queue problem. Stochastic Models 6: 691-713.
4. Adan, I.J.B.F., Wessels, J., and Zijm, W.H.M. (1991). Analysis of the asymmetric shortest queue problem. Queueing Systems 8: 1-58.
5. Adan, I.J.B.F., Wessels, J., and Zijm, W.H.M. (1993). Analysing multiprogramming queues by generating functions. SIAM Journal of Applied Mathematics 53: 1123-1131.
6. Adan, I.J.B.F., Wessels, J., and Zijm, W.H.M. (1993). A compensation approach for two-dimensional Markov processes. Advances in Applied Probability 25: 783-817.
7. Cohen, J.W. (1994). On a class of two-dimensional nearest neighbouring random walks. Journal of Applied Probability 31A (to appear).

8. Cohen, J.W., and Boxma, O.J. (1983). Boundary Value Problems in Queueing System Analysis, North-Holland, Amsterdam.
9. Flatto, L. and McKean, H.P. (1977). Two queues in parallel. Communications on Pure and Applied Mathematics 30: 255-263.
10. Hofri, M. (1978). A generating-function analysis of multiprogramming queues. International Journal of Computer and Informational Sciences 7: 121-155.
11. Jaffe, S. (1989). Equilibrium results for a pair of coupled discrete-time queues. Ultracomputer Note, NYA Ultracomputer Research Lab, Courant Institute of Mathematical Sciences, New York.
12. Jaffe, S. (1992). The equilibrium distribution for a clocked buffered switch. Probability in the Engineering and Informational Sciences 6: 425-438.
13. Kingman, J.F.C. (1961). Two similar queues in parallel. Annals of Mathematical Statistics 32: 1314-1323.
14. Malyshev, V.A. (1972). Classification of two-dimensional positive random walks and almost linear semimartingales. Doklady Akademii Nauk SSSR 202: 526-528 (English translation in Soviet Mathematics Doklady 13).

Routing with Breakdowns

Isi Mitrani*
Computing Science Dept., University of Newcastle,
Newcastle upon Tyne NE1 7RU, England

Paul E. Wright
Mathematical Sciences Center, AT&T Bell Laboratories,
Murray Hill, NJ 07974-0636, U.S.A.

Jobs generated by a single Poisson source can be routed through N alternative gateways, modelled as parallel $M/M/1$ queues. The servers at those queues are subject to random breakdowns and repairs. When a breakdown occurs, all jobs present in the corresponding queue are lost; moreover, no incoming jobs are directed to that queue during the subsequent repair period. The marginal queue size distributions are determined by finding the roots of a polynomial inside the unit disc, and solving a set of simultaneous linear equations. In the case $N = 2$, it is also possible to find the joint equilibrium distribution of the numbers of jobs in the two queues, by a reduction to a Dirichlet boundary value problem on a circle.

1 INTRODUCTION

The analysis of queueing systems where servers are subject to random breakdowns and repairs has long been recognised as an interesting and important research topic. The literature on the subject is quite extensive, and the applications are many and varied. However, the great majority of the work has concentrated on models involving a single job queue served by one or more processors (e.g., see [1, 12, 14, 15]). Very few results are available for systems with more than one queue. An approximate solution for open and closed networks with unreliable servers was suggested by Mitrani [11] while Mikou [10] analysed exactly a tightly coupled two-node network with simultaneous breakdowns and repairs. That analysis provides ample illustration of the formidable mathematical difficulties posed by this kind of model.

---

*Partially funded by the European Community, through the Basic Research projects QMIPS (Quantitative Methods In Parallel Systems), and PDCS II (Predictably Dependable Computer Systems). The support of AT&T Bell Laboratories, which enabled the first author to visit the second, is also gratefully acknowledged.

Here we consider a system where an incoming stream of jobs is split into N substreams, each of which is directed to a separate server and its associated queue. The service, breakdown and repair processes at the different servers are independent of each other and have different parameters, in general. The occurrence of a breakdown at server k has two consequences: (i) all jobs that are currently in queue k are lost, and (ii) all arrivals from substream k during the ensuing inoperative period are redirected to other servers if any are operative; otherwise they too are lost. This redirection implies dependencies between the queues.

Our motivation for studying this system comes from the field of networking: the jobs are messages generated by some source, and the servers are alternative gateways through which those messages may be routed. Gateways are subject to failures that are sufficiently catastrophic to lose all messages currently accumulated. The source finds out about such failures and, if possible, redirects traffic. The assumption that incoming jobs are lost when all servers are broken could be restated by saying that the source ceases to generate messages during those periods.

The model and its parameters are specified in section 2. The object of greatest theoretical importance in this connection is the joint stationary distribution of the set of operative servers and the numbers of jobs in the corresponding queues. To find that distribution one has to solve a non-separable multidimensional Markov process, which is an intractable problem in the general case. However, we are able to determine the joint distribution in the case $N = 2$, by a reduction to a Dirichlet boundary value problem on a circle (section 5).

On the other hand, the performance measures of practical interest are mainly concerned with averages, e.g. the average number of jobs present at a given server or the total average number of jobs lost per unit time. To calculate such performance measures, it is enough to determine the marginal queue size distributions. This last problem can be solved, at least in principle, for arbitrary N (section 3). In particular, the ability to compute mean queue sizes implies that one can tackle various optimization problems. For instance, how should the input stream of jobs be split among the possible routes in order to minimize the total average number of losses per unit time? This last problem is examined numerically in section 4, for $N = 2$.

2 THE MODEL

Jobs arrive into the system in a Poisson stream with rate λ. There are N servers, each with an associated unbounded queue, to which incoming jobs may be directed. Server i goes through alternating independent operative and inoperative periods, distributed exponentially with means $1/\xi_i$ and $1/\eta_i$, respectively. While it is operative, the jobs in its queue receive exponentially distributed services with mean $1/\mu_i$, and depart upon completion. When a server becomes inoperative (breaks down), the corresponding queue is cleared and all jobs in it, whether waiting or in service, are lost.

Server i is assigned a positive 'routing weight', w_i. If, when a new job arrives into the system, the set of operative servers, σ, is non-empty ($\sigma \subset$

$\{1, 2, \ldots, N\}$), then the job is placed into one of the queues in σ, choosing queue i with probability $q_i(\sigma)$, given by

$$q_i(\sigma) = \frac{w_i}{\sum_{j \in \sigma} w_j} \quad , \quad i \in \sigma \; .$$

These routing decisions are independent of each other and of the numbers of jobs in queues.

If all servers are broken down at an arrival instant, then the incoming job is lost.

The system state at time t is described by the pair $X(t) = [\mathbf{I}(t)\, , \mathbf{J}(t)]$, where $\mathbf{I}(t) \subset \{1, 2, \ldots, N\}$ is the set of currently operative servers (represented, if not empty, as a vector of indices), and $\mathbf{J}(t)$ is the vector of corresponding queue sizes. More precisely, if $\mathbf{I}(t) = (i_1, i_2, \ldots, i_k)$, then $\mathbf{J}(t)$ has k elements, the first of which is the number of jobs in queue i_1, the second is the number of jobs in queue i_2, etc. Remember that the queues of inoperative servers are necessarily empty, and so their sizes need not be specified. If $\mathbf{I}(t)$ is the empty set, $\emptyset$, then $\mathbf{J}(t)$ is absent.

The above assumptions ensure that $\{X(t)\, , \, t \geq 0\}$ is an irreducible Markov process. Moreover, it is intuitively obvious that, as long as $\xi_i > 0$ and $\eta_i < \infty$ for all i, then no matter how heavy the traffic, the process is ergodic. This is due to the fact that queues are emptied from time to time as result of server breakdowns, and so their sizes cannot drift to infinity. Define the equilibrium probabilities

$$p_\sigma(\mathbf{n}) = \lim_{t \to \infty} P[\mathbf{I}(t) = \sigma \, , \, \mathbf{J}(t) = \mathbf{n}] \; , \tag{1}$$

where σ can be any subset of $\{1, 2, \ldots, N\}$ (including $\emptyset$ and the entire set), and $\mathbf{n}$ is a non-negative integer vector with $|\sigma|$ elements ($|\sigma|$ is the cardinality of σ; if $\sigma = \emptyset$, then $\mathbf{n}$ is absent).

It is convenient to introduce some shorthand notation for constructing new states from old ones. Thus, if $(\sigma, \mathbf{n})$ is a system state and $i \in \sigma$, then $(\sigma, \mathbf{n} + \mathbf{1}_i)$ is the state in which queue i has one more job; $(\sigma, \mathbf{n} - \mathbf{1}_i)$ is the state in which queue i has one less job; $(\sigma - \{i\}, \mathbf{n})$ is the state in which server i is inoperative and queue i is empty; all other servers and queues are as before (the vector $\mathbf{n}$ has one less element, since queue i is not included). If $i \in \overline{\sigma}$, where $\overline{\sigma}$ is the complement of σ with respect to $\{1, 2, \ldots, N\}$, then $(\sigma + \{i\}, (\mathbf{n}, m))$ is the state where an additional server, i, is operative and the corresponding queue has m jobs.

Also, let δ_B be the indicator function of the condition B:

$$\delta_B = \begin{cases} 1 \text{ if } B \text{ holds,} \\ 0 \text{ otherwise.} \end{cases}$$

Now we can write a set of balance equations for the equilibrium probabilities:

$$\left[\lambda \delta_{(\sigma \neq \emptyset)} + \sum_{i \in \sigma} \mu_i \delta_{(n_i > 0)} + \sum_{i \in \sigma} \xi_i + \sum_{i \in \overline{\sigma}} \eta_i \right] p_\sigma(\mathbf{n}) =$$

$$\sum_{i\in\sigma}\lambda q_i(\sigma)p_\sigma(\mathbf{n}-\mathbf{1}_i)+\sum_{i\in\sigma}\mu_i p_\sigma(\mathbf{n}+\mathbf{1}_i)+\sum_{i\in\sigma}\delta_{(n_i=0)}\eta_i p_{\sigma-\{i\}}(\mathbf{n})+$$

$$\sum_{i\in\overline{\sigma}}\xi_i\sum_{m=0}^{\infty}p_{\sigma+\{i\}}(\mathbf{n},m)\ \ ,\sigma\subset\{1,2,\ldots,N\}\ ,\ \mathbf{n}\geq\mathbf{0}\ . \tag{2}$$

Here, an empty sum is equal to 0 by definition, as are the probabilities of states with negative queue sizes or with more than N operative processors. In addition to the above, a normalizing equation must be satisfied:

$$\sum_{\sigma\subset\{1,2,\ldots,N\}}\ \sum_{\mathbf{n}\geq\mathbf{0}}p_\sigma(\mathbf{n})=1\ . \tag{3}$$

In theory, the equilibrium distribution (1) is uniquely determined by equations (2) and (3) (see [4]). In practice, solving those equations is far from trivial even when $N=2$ (see section 5), and is an open problem for $N>2$. On the other hand, the marginal distributions of isolated queues can be determined without too much difficulty for larger values of N, as described in the next section. The marginal probability, p_σ, that the servers in σ are operative and the ones in $\overline{\sigma}$ are broken, is of course given by

$$p_\sigma=\prod_{i\in\sigma}\frac{\eta_i}{\xi_i+\eta_i}\prod_{i\in\overline{\sigma}}\frac{\xi_i}{\xi_i+\eta_i}\ \ ,\ \ \sigma\subset\{1,2,\ldots,N\}\ , \tag{4}$$

where an empty product is by definition equal to 1. These expressions follow from the fact that servers break down and are repaired independently of each other; they can also be established by summing (2) over all $\mathbf{n}\geq\mathbf{0}$, and then verifying that (4) satisfies the resulting equations.

3 MARGINAL DISTRIBUTIONS

Consider, without loss of generality, the marginal distribution of queue 1. Given a subset, $\sigma\subset\{1,2,\ldots,N\}$, of which 1 is a member, let $p_\sigma(n,\cdot)$ be the equilibrium probability that the servers in σ are operative, the ones in $\overline{\sigma}$ are broken down, and there are n jobs in queue 1. For every value of n, there are 2^{N-1} such probabilities (corresponding to the subsets that include 1), and their sum is equal to the equilibrium probability that there are n jobs in queue 1 and server 1 is operative.

By summing the equations in (2) over all vectors $\mathbf{n}$ in which $n_1=n$, and performing the appropriate cancellations, we get

$$\left[\lambda q_1(\sigma)+\mu_1\delta_{(n_1>0)}+\sum_{i\in\sigma}\xi_i+\sum_{i\in\overline{\sigma}}\eta_i\right]p_\sigma(n,\cdot)=$$

$$\lambda q_1(\sigma)p_\sigma(n-1,\cdot)+\mu_1 p_\sigma(n+1,\cdot)+\eta_1\delta_{(n_1=0)}p_{\sigma-\{1\}}+$$

$$\sum_{i\in\sigma-\{1\}}\eta_i p_{\sigma-\{i\}}(n,\cdot)+\sum_{i\in\overline{\sigma}}\xi_i p_{\sigma+\{i\}}(n,\cdot)\ \ ,\ \ n=0,1,\ldots\ . \tag{5}$$

The solution of equations (5) is best achieved by introducing the generating functions

$$g_\sigma(z) = \sum_{n=0}^{\infty} p_\sigma(n,\cdot)z^n \quad , \quad 1 \in \sigma \subset \{1,2,\dots,N\} \, . \tag{6}$$

Knowledge of the functions (6), together with the fact that queue 1 is empty when server 1 is inoperative, would yield the generating function of the marginal distribution of queue 1, $h_1(z)$:

$$h_1(z) = \frac{\xi_1}{\xi_1 + \eta_1} + \sum_{1 \in \sigma \subset \{1,2,\dots,N\}} g_\sigma(z) \, . \tag{7}$$

A similar development would lead to the generating function of the marginal distribution of queue i, $h_i(z)$, for $i = 2,3,\dots,N$. Various performance measures can then be obtained. For example, the total average number of jobs that are lost per unit time, L, is given by

$$L = \lambda p_{\{\emptyset\}} + \sum_{i=1}^{N} \xi_i h_i'(1) \, . \tag{8}$$

Another quantity of interest is the average number, m_i, of successful jobs in queue i. A successful job is one that manages to complete its service, rather than being lost as a result of a breakdown. Note that, if there are n jobs present in queue i, then the probability that j of them are successful is equal to $[\mu_i/(\mu_i + \xi_i)]^j[\xi_i/(\mu_i + \xi_i)]$ for $j = 0,1,\dots,n-1$, and to $[\mu_i/(\mu_i + \xi_i)]^n$ for $j = n$. From this we can derive a simple relation between the generating function of the number of successful jobs in queue i, $s_i(z)$, and the marginal generating function $h_i(z)$:

$$s_i(z) = \frac{\mu_i(1-z)}{\xi_i + \mu_i(1-z)} h_i\left(\frac{\mu_i z}{\mu_i + \xi_i}\right) + \frac{\xi_i}{\xi_i + \mu_i(1-z)} \, .$$

Hence, the average number of successful jobs in queue i is given by

$$m_i = s_i'(1) = \frac{\mu_i}{\xi_i}\left[1 - h_i\left(\frac{\mu_i}{\mu_i + \xi_i}\right)\right] \, . \tag{9}$$

The overall average response time, W, of a successful job, is obtained by an appeal to Little's theorem:

$$W = \frac{1}{\lambda - L}\sum_{i=1}^{N} m_i \, . \tag{10}$$

So, the problem is to determine $g_\sigma(z)$. Multiplying (5) by z^n and summing over all $n \geq 0$ yields, after some manipulations,

$$\left[\lambda q_1(\sigma)z(1-z) - \mu_1(1-z) + \sum_{i\in\sigma} \xi_i z + \sum_{i\in\overline{\sigma}} \eta_i z\right] g_\sigma(z)$$

$$- \sum_{i \in \sigma - \{1\}} \eta_i z g_{\sigma - \{i\}}(z) - \sum_{i \in \overline{\sigma}} \xi_i z g_{\sigma + \{i\}}(z) =$$

$$\eta_1 z p_{\sigma - \{1\}} - \mu_1 (1 - z) g_\sigma(0) \ \ , 1 \in \sigma \subset \{1, 2, \ldots, N\} \, . \tag{11}$$

Here we have 2^{N-1} linear equations for the 2^{N-1} generating functions (6). Unfortunately, the right-hand sides of (11) contain not only the known constants $p_{\sigma-\{1\}}$ (given by (4)), but also the 2^{N-1} unknown probabilities, $g_\sigma(0)$. To show how the latter can be determined, it is helpful to re-write (11) in matrix form:

$$A(z)\mathbf{g}(z) = \mathbf{b}(z) \, , \tag{12}$$

where the elements of $\mathbf{g}(z)$ are the generating functions $g_\sigma(z)$, those of $\mathbf{b}(z)$ are the right-hand sides of (11), $b_\sigma(z) = \eta_1 z p_{\sigma-\{1\}} - \mu_1(1-z)g_\sigma(0)$, and $A(z)$ is the $2^{N-1} \times 2^{N-1}$ coefficient matrix. Then the solution is given by Cramer's rule [6]:

$$g_\sigma(z) = \frac{D_\sigma(z)}{D(z)} \ , \ 1 \in \sigma \subset \{1, 2, \ldots, N\} \, , \tag{13}$$

where $D(z)$ is the determinant of $A(z)$ and $D_\sigma(z)$ is the determinant of the matrix obtained from $A(z)$ by replacing its σ-column with the vector $\mathbf{b}(z)$.

Now note that $g_\sigma(z)$, being probability generating functions, are analytic in the interior of the unit circle, $|z| < 1$. Therefore, if $D(z)$, which is a polynomial of degree 2^N, has a root in that region, say z_0, then $D_\sigma(z)$ must vanish at z_0 for all σ under consideration. A requirement of the form $D_\sigma(z_0) = 0$ provides a linear equation for the unknown probabilities. Moreover, if z_0 is a root of multiplicity k, then $k-1$ derivatives of $D_\sigma(z)$ must vanish at z_0, yielding $k-1$ further equations.

On the other hand, since the columns of $A(z_0)$ are linearly dependent, it follows that if $D_\sigma(z)$, or one of its derivatives, vanishes at z_0 for a particular index σ, then it does so for all other indices. In other words, if one of the generating functions in (6) is analytic at z_0, then they all are.

Thus, every root of $D(z)$ in the unit disc provides as many linear equations involving the unknown probabilities, as is its multiplicity (a pair of complex conjugate roots provides two equations with real coefficients). For the purpose of these equations, σ can be fixed arbitrarily, e.g. $\sigma = \{1\}$. Consequently, the determination the 2^{N-1} unknowns is made possible by the following:

PROPOSITION 1 *The polynomial $D(z)$ has exactly 2^{N-1} roots in the unit disc, assuming that each root is counted according to its multiplicity.*

A proof of this proposition, in the case of $N = 2$, is presented later in this section. We have been unable to prove the result for arbitrary N. However, an intuitive argument for the general validity of the proposition can be based on the fact that the balance and normalizing equations (2) and (3) have a unique solution. Indeed, if $D(z)$ had less that 2^{N-1} roots in the unit disc, then it would be possible to construct more than one solution, whereas if it had more than 2^{N-1} roots, the equations would be overdetermined and there would be no solution. We have carried out a few numerical experiments with $N = 3$, and in all cases have observed the 4 roots in the unit disc predicted by the proposition.

Let us examine in greater detail the system with two queues. There are now two marginal generating functions involving queue 1, $g_{\{1\}}(z)$ and $g_{\{1,2\}}(z)$. Routing decisions for incoming jobs are made only when both servers are operative. To simplify the notation, let $\lambda q_i(\{1,2\}) = \lambda_i$, $i = 1, 2$.

The set of equations (12) has the form

$$\begin{bmatrix} a(z) & -\xi_2 z \\ -\eta_2 z & b(z) \end{bmatrix} \begin{bmatrix} g_{\{1\}}(z) \\ g_{\{1,2\}}(z) \end{bmatrix} = \begin{bmatrix} \eta_1 z p_{\{\emptyset\}} - \mu_1(1-z) g_{\{1\}}(0) \\ \eta_1 z p_{\{2\}} - \mu_1(1-z) g_{\{1,2\}}(0) \end{bmatrix}, \quad (14)$$

where

$$a(z) = \lambda z(1-z) - \mu_1(1-z) + \xi_1 z + \eta_2 z \ ,$$

$$b(z) = \lambda_1 z(1-z) - \mu_1(1-z) + \xi_1 z + \xi_2 z \ .$$

The solution of (14) is

$$g_{\{1\}}(z) = \frac{1}{D(z)} det \begin{bmatrix} \eta_1 z p_{\{\emptyset\}} - \mu_1(1-z) g_{\{1\}}(0) & -\xi_2 z \\ \eta_1 z p_{\{2\}} - \mu_1(1-z) g_{\{1,2\}}(0) & b(z) \end{bmatrix},$$

$$g_{\{1,2\}}(z) = \frac{1}{D(z)} det \begin{bmatrix} a(z) & \eta_1 z p_{\{\emptyset\}} - \mu_1(1-z) g_{\{1\}}(0) \\ -\eta_2 z & \eta_1 z p_{\{2\}} - \mu_1(1-z) g_{\{1,2\}}(0) \end{bmatrix}, \quad (15)$$

where

$$D(z) = a(z)b(z) - \xi_2 \eta_2 z^2 \ . \quad (16)$$

When both λ_1 and λ_2 are non-zero, the determinant $D(z)$ is a polynomial of degree 4. Denote its four roots by α_1, α_2, α_3, α_4. According to Proposition 1, two of them are inside the unit disc and two are outside. In fact, a slightly stronger result can be proved:

LEMMA 1 *The roots α_k are real and distinct, and if numbered in ascending order satisfy the inequalities* $0 < \alpha_1 < \alpha_2 < 1 < \alpha_3 < \alpha_4$.

Proof. From the definition of $a(z)$ it follows immediately that $a(0) < 0$, $a(1) > 0$ and $a(\infty) < 0$. Hence, $a(z)$, which is quadratic, has one root, ζ_1, in the interval $(0,1)$, and one root, ζ_2, in the interval $(1,\infty)$. This, together with the definition of $D(z)$, implies that $D(0) > 0$, $D(\zeta_1) < 0$, $D(1) > 0$, $D(\zeta_2) < 0$ and $D(\infty) > 0$. Therefore, $D(z)$ has a real root in each of the intervals $(0,\zeta_1)$, $(\zeta_1, 1)$, $(1, \zeta_2)$ and (ζ_2, ∞), q.e.d.

The requirement that the numerators in the right-hand sides of (15) should vanish at points α_1 and α_2, yields four linear equations for $g_{\{1\}}(0)$ and $g_{\{1,2\}}(0)$, of which two are independent. It is not too difficult to show that the solution of those equations, combined with expressions (4), is

$$g_{\{1\}}(0) = \frac{\xi_1 \xi_2 \eta_1 \alpha_1 \alpha_2 [\lambda_1(1 + \alpha_1\alpha_2 - \alpha_1 - \alpha_2) + \xi_1 + \xi_2 + \eta_2]}{\mu_1(\xi_1 + \eta_1)(\xi_2 + \eta_2)(1 - \alpha_1)(1 - \alpha_2)(\mu_1 - \lambda_1\alpha_1\alpha_2)} ,$$

$$g_{\{1,2\}}(0) = \frac{\xi_1 \eta_1 \eta_2 \alpha_1 \alpha_2 [\lambda(1 + \alpha_1\alpha_2 - \alpha_1 - \alpha_2) + \xi_1 + \xi_2 + \eta_2]}{\mu_1(\xi_1 + \eta_1)(\xi_2 + \eta_2)(1 - \alpha_1)(1 - \alpha_2)(\mu_1 - \lambda\alpha_1\alpha_2)} . \tag{17}$$

We have established that $g_{\{1\}}(z)$ and $g_{\{1,2\}}(z)$, which are rational functions, have exactly two poles, α_3 and α_4. Moreover, both these functions vanish at infinity, since the numerators in (15) are of lower degree than the denominators. Therefore, we may write

$$g_{\{1\}}(z) = \frac{c_1}{\alpha_3 - z} + \frac{c_2}{\alpha_4 - z} , \quad g_{\{1,2\}}(z) = \frac{c_3}{\alpha_3 - z} + \frac{c_4}{\alpha_4 - z} , \tag{18}$$

where c_1, c_2, c_3, c_4 are some real constants. These constants can be determined easily as the residues of (15) at α_3 and α_4. For example, multiplying the first equation in (15) by $\alpha_3 - z$ and then setting $z = \alpha_3$, we get

$$c_1 = -\frac{det \begin{bmatrix} \eta_1\alpha_3 p_{\{\emptyset\}} - \mu_1(1-\alpha_3)g_{\{1\}}(0) & -\xi_2\alpha_3 \\ \eta_1\alpha_3 p_{\{2\}} - \mu_1(1-\alpha_3)g_{\{1,2\}}(0) & b(\alpha_3) \end{bmatrix}}{\lambda\lambda_1(\alpha_3 - \alpha_1)(\alpha_3 - \alpha_2)(\alpha_3 - \alpha_4)} .$$

From the above arguments we conclude that the marginal distribution of queue 1 is a mixture of two geometric distributions, with parameters $1/\alpha_3$ and $1/\alpha_4$ respectively. More precisely, the probability that there are n jobs in queue 1, $p_1(n)$, is given by

$$p_1(n) = \frac{c_1 + c_3}{\alpha_3}\alpha_3^{-n} + \frac{c_2 + c_4}{\alpha_4}\alpha_4^{-n} \quad , \quad n = 1, 2, \ldots .$$

The probability $p_1(0)$ is determined from the normalizing equation.

A similar result is available for queue 2. The developments are exactly the same, up to an exchange of server indices. The roots α_k are replaced by the roots β_k of a determinant analogous to $D(z)$; again, two of those roots are in the interval $(0, 1)$, and the other two are in the interval $(1, \infty)$.

If $\lambda_1 = 0$ (i.e., jobs are sent to server 1 only when server 2 is inoperative), then the above solution is subject to a small modification. The factor $b(z)$ which appears in (16) loses its quadratic term and becomes linear. The determinant is then a polynomial of degree 3 and the root α_4 does not exist. Also, in the second equation in (15), the numerator and the denominator are of the same degree, which means that the function $g_{\{1,2\}}(z)$ does not vanish at infinity. In that case, equations (18) take the form

$$g_{\{1\}}(z) = \frac{c_1}{\alpha_3 - z} \quad , \quad g_{\{1,2\}}(z) = \frac{\eta_1 p_{\{2\}} + \mu_1 g_{\{1,2\}}(0)}{\mu_1 + \xi_1 + \xi_2} + \frac{c_3}{\alpha_3 - z} .$$

Similar remarks apply when $\lambda_2 = 0$.

4 NUMERICAL RESULTS

We wish to examine empirically the effect of routing decisions on performance, in the context of a system with two queues. The splitting of the input stream among the servers is now governed by a single parameter, q. When both servers are operative, a fraction q of the incoming jobs is directed to server 1 and the rest to server 2. Thus $\lambda_1 = \lambda q$ and $\lambda_2 = \lambda(1-q)$. The performance measures of interest are the average rate of loss, L, given by (8), and the average response time of a successful job, W, obtained from (9) and (10). With the other parameters fixed, these performance measures are regarded as functions of q on the interval $0 \leq q \leq 1$.

The solution described in the previous section was implemented in order to compute L and W (with the appropriate modifications applied at the extreme points, $q = 0$ and $q = 1$). Figure 1 shows four plots of the loss rate as a function of q. The fixed parameter values were $\lambda = 1.8$, $\mu_1 = \mu_2 = 1$, $\xi_1 = 0.01$, $\eta_1 = \eta_2 = 0.1$. The four curves correspond to different values of ξ_2: 0.01, 0.05, 0.1 and 0.5. Note that, in the absence of breakdowns, this system would be ergodic only when $0.8/1.8 < q < 1/1.8$.

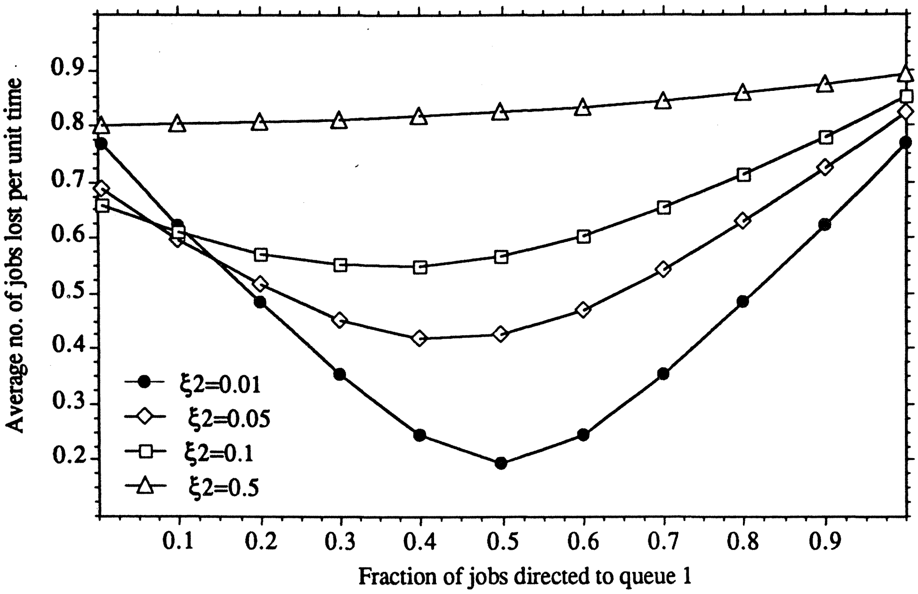


Figure 1. Average rate of loss as a function of q: $\lambda = 1.8$, $\mu_1 = \mu_2 = 1$, $\xi_1 = 0.01$, $\eta_1 = \eta_2 = 0.1$

In the symmetric case, $\xi_2 = 0.01$ (the two servers are identical), the figure confirms our expectations, namely that in order to minimize the loss rate, the input stream should be split equally among the two queues. However, when the breakdown rate of server 2 increases, the results become counter-intuitive. The plots suggest that it is better to direct a larger fraction of the incoming jobs to the server with the larger breakdown rate. Indeed, sending all jobs to server 2 (when both are operative) leads to a lower loss rate when $\xi_2 = 0.05$

than when $\xi_2 = 0.01$.

The explanation of this phenomenon lies in the transient behaviour of the queues. When $\lambda_i > \mu_i$, queue i tends to just grow during the operative periods of its server; if the latter are long, then the ensuing breakdowns cause many job losses. Thus, slightly more frequent breakdowns may in fact be beneficial in reducing queue size growth and job losses. Hence it may be better to direct more jobs to the server that breaks down more frequently. It should be remembered that, regardless of the value of q, when one server is broken, all incoming jobs are directed to the other server.

For the same parameter sets, figure 2 shows the behaviour of the average response time of successful jobs.

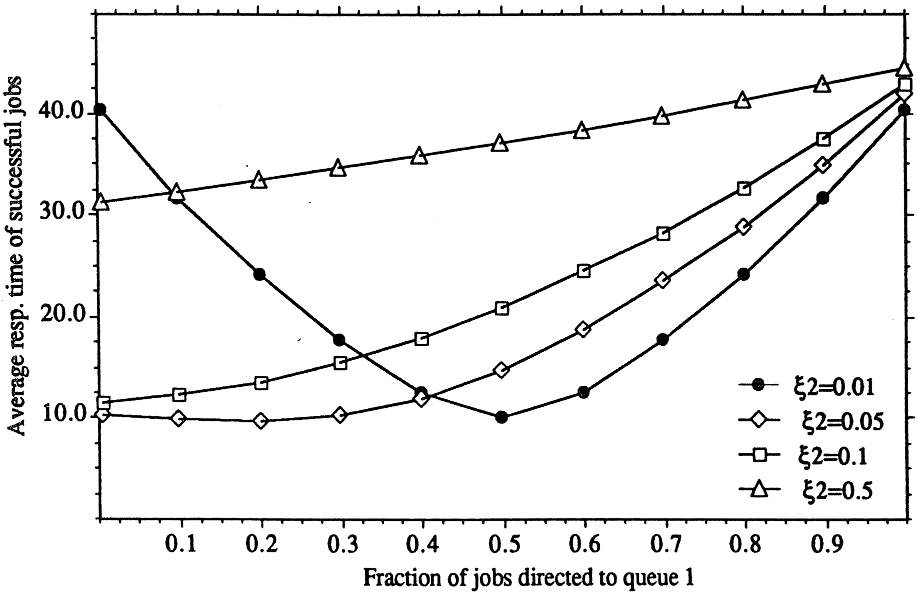


Figure 2. Average response time of successful jobs as a function of q: $\lambda = 1.8$, $\mu_1 = \mu_2 = 1$, $\xi_1 = 0.01$, $\eta_1 = \eta_2 = 0.1$

Again, in the symmetric case, the optimal splitting is obtained for $q = 0.5$. However, the effect of growing queues is even more pronounced here. Already when $\xi_2 = 0.1$, it is best to direct all jobs to server 2, given that both servers are operative. The explanation is the same as above.

5 THE JOINT QUEUE SIZE DISTRIBUTION FOR $N = 2$

When there are two gateways available, the possible subsets, σ, of operative servers, and the associated equilibrium state probabilities, as defined in (1), are:

$$p_\emptyset,\ p_{\{1\}}(n),\ p_{\{2\}}(n),\ p_{\{1,2\}}(n_1, n_2)\ .$$

The generating function of $p_{\{1\}}(n)$, $g_{\{1\}}(z)$, is given by the first equation in (18). A similar expression, involving roots β_3 and β_4, exists for the generating function of $p_{\{2\}}(n)$, $g_{\{2\}}(z)$. The problem facing us now is to determine the bivariate generating function

$$g(x,y) = \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} p_{\{1,2\}}(n_1,n_2) x^{n_1} y^{n_2} \ . \tag{19}$$

We shall use the notation λ_1 and λ_2 introduced in the last section. Multiplying equations (2), with $\sigma = \{1,2\}$, by $x^{n_1}y^{n_2}$ and summing, leads to

$$R(x,y)g(x,y) = \mu_2 x(y-1)g(x,0) + \mu_1 y(x-1)g(0,y) +$$
$$\eta_2 xy g_{\{1\}}(x) + \eta_1 xy g_{\{2\}}(y) \ , \tag{20}$$

where

$$R(x,y) = (\lambda_1 x - \mu_1)(1-x)y + (\lambda_2 y - \mu_2)(1-y)x + (\xi_1 + \xi_2)xy \ .$$

To solve this equation, it is necessary to find the two unknown functions, $g(x,0)$ and $g(0,y)$. A 'short cut' to the solution does not seem to exist. For example, one might have hoped that when both servers are operative, the two queues are independent of each other. Then the solution would be of the form $g(x,y) = g(x,1)g(1,y)/g(1,1)$. Unfortunately, a direct substitution of (18) into (20) shows that this cannot be true.

The determination of $g(x,0)$ and $g(0,y)$ is based on the observation that the right-hand side of (20) vanishes wherever $g(x,y)$ is finite and $R(x,y)$ is 0. In particular, (18) implies that $g(x,1)$ is analytic in the interior of the disc $|x| < \alpha_3$, which shows that $g(x,0)$ is also analytic in that region. Similarly, $g(1,y)$ and $g(0,y)$ are analytic in the interior of the disc $|y| < \beta_3$. Hence, $g(x,y)$ is analytic in the polydisc $|x| < \alpha_3$, $|y| < \beta_3$. Therefore, the equation

$$\mu_2 x(y-1)g(x,0) + \mu_1 y(x-1)g(0,y) + \eta_2 xy g_1(x) + \eta_1 xy g_2(y) = 0 \ , \tag{21}$$

is valid on the set of points $\{(x,y) \mid R(x,y) = 0, \ |x| < \alpha_3, \ |y| < \beta_3\}$.

Much of what follows is similar to the solution of another two-dimensional Markov process in Fayolle and Iasnogorodski [2]. We shall miss some of the details of the development; the interested reader is directed to [2] for a fuller treatment. First, it is important to examine more closely the algebraic curve defined by $R(x,y) = 0$. On that curve, we can regard either y as a function of x, or x as a function of y. Solving $R(x,y) = 0$ for y gives

$$y(x) = \frac{\lambda_1 x^2 - \gamma x + \mu_1 \pm \sqrt{(\lambda_1 x^2 - \gamma x + \mu_1)^2 - 4\lambda_2 \mu_2 x^2}}{2\lambda_2 x} \ , \tag{22}$$

where $\gamma = \lambda_1 + \lambda_2 + \mu_1 + \mu_2 + \xi_1 + \xi_2$.

The following simple properties of the above will be needed:

(a) The two-valued algebraic function $y(x)$ has four real branch points which satisfy the inequalities $0 < x_1 < x_2 < 1 < x_3 < x_4$.

(b) Each of the segments $[x_1, x_2]$ and $[x_3, x_4]$ is mapped by $y(x)$ onto the circle $|y| = \sqrt{\mu_2/\lambda_2}$.

To prove (a), factorize the discriminant in (22) into a product of two quadratic polynomials, $(\lambda_1 x^2 - \gamma x + \mu_1 + 2\sqrt{\lambda_2\mu_2 x})(\lambda_1 x^2 - \gamma x + \mu_1 - 2\sqrt{\lambda_2\mu_2 x})$; it is easily seen that each of the factors changes sign between 0 and 1, and between 1 and ∞. Property (b) follows from the fact that the product of the two branches of $y(x)$ is constant: $y_1(x)y_2(x) = \mu_2/\lambda_2$. On the segments $[x_1, x_2]$ and $[x_3, x_4]$, those branches are complex conjugate and hence of equal modulus; the latter must therefore be equal to $\sqrt{\mu_2/\lambda_2}$.

Similarly, if $R(x, y) = 0$ is solved for x, then (a) the two-valued function $x(y)$ has four real branch points, $0 < y_1 < y_2 < 1 < y_3 < y_4$; (b) each of the segments $[y_1, y_2]$ and $[y_3, y_4]$ is mapped by $x(y)$ onto the circle $|x| = \sqrt{\mu_1/\lambda_1}$.

It can also be shown that a branch of $y(x)$ maps the circle $|x| = \sqrt{\mu_1/\lambda_1}$ onto the segment $[y_1, y_2]$, sweeping the latter twice. A branch of $x(y)$ maps the circle $|y| = \sqrt{\mu_2/\lambda_2}$ onto the segment $[x_1, x_2]$, sweeping the latter twice.

When determining the function $g(x, 0)$, there are two main cases to consider.

(i) $\sqrt{\mu_1/\lambda_1} < \alpha_3$.

In this case, $g(x, 0)$ and $g_{\{1\}}(x)$ are analytic in the closed disc $|x| \le \sqrt{\mu_1/\lambda_1}$. Consider the pairs (x, y), where x is on the circle $|x| = \sqrt{\mu_1/\lambda_1}$, $y = y(x)$ is on the segment $[y_1, y_2]$, and $R(x, y) = 0$. The mapping $y(x)$ is given by (22), with a minus sign in front of the square root. For all such points, equation (21) holds.

Multiply (21) by $(\bar{x} - 1)$, where $\bar{x}$ is the complex conjugate of x, and observe that the term $\mu_1 y(x-1)(\bar{x}-1)g(0, y)$ is real. Taking imaginary parts and using the fact that $x\bar{x} = \mu_1/\lambda_1$, yields

$$Im[(\frac{\mu_1}{\lambda_1} - x)g(x, 0)] = u(x) \quad , \quad |x| = \sqrt{\frac{\mu_1}{\lambda_1}} \; , \tag{23}$$

where

$$u(x) = \frac{y(x)}{\mu_2[1 - y(x)]} Im\{(\frac{\mu_1}{\lambda_1} - x)[\eta_2 g_{\{1\}}(x) + \eta_1 g_{\{2\}}(y(x))]\} \; .$$

We have arrived at the following problem: Find a function which is analytic in the interior of a closed contour (a circle), continuous on the boundary, and satisfying on that contour a condition of the form (23). This is a Dirichlet boundary value problem and its solution is known (see, for example, [13, 5]). Passing to polar coordinates by the change of variables $x = \sqrt{\mu_1/\lambda_1} e^{\imath\vartheta}$, and applying Schwartz's formula, we get an expression for $g(z, 0)$ in the interior of the disc:

$$\left(\sqrt{\frac{\mu_1}{\lambda_1}} - z\right) g\left(\sqrt{\frac{\mu_1}{\lambda_1}} z, 0\right) = \frac{\imath}{2\pi}\int_0^{2\pi} v(\vartheta)\frac{e^{\imath\vartheta} + z}{e^{\imath\vartheta} - z} d\vartheta + C \; , \; |z| < 1, \tag{24}$$

where $v(\vartheta) = \sqrt{\lambda_1/\mu_1} u[\sqrt{\mu_1/\lambda_1} e^{\imath\vartheta}]$ and C is a constant. Equation (24) may be rewritten as

$$\left(\sqrt{\frac{\mu_1}{\lambda_1}} - z\right) g\left(\sqrt{\frac{\mu_1}{\lambda_1}} z, 0\right) =$$
$$\frac{\imath}{2\pi}\int_{-\pi}^{\pi} \frac{1 - z^2 - 2\imath z \sin\vartheta}{1 - 2z\cos\vartheta + z^2} v(\vartheta)\, d\vartheta + C \quad , \quad |z| < 1 \; . \tag{25}$$

Observing now that $v(\vartheta) = -v(-\vartheta)$, we obtain finally,

$$\left(\sqrt{\frac{\mu_1}{\lambda_1}} - z\right) g\left(\sqrt{\frac{\mu_1}{\lambda_1}} z, 0\right) =$$
$$\frac{2}{\pi}\int_{0}^{\pi} \frac{z \sin\vartheta}{1 - 2z\cos\vartheta + z^2} v(\vartheta)\, d\vartheta + C \quad , \quad |z| < 1 \; . \tag{26}$$

It remains to determine the constant C. If $\sqrt{\mu_1/\lambda_1} \leq 1$, then this is done by setting $z = \sqrt{\mu_1/\lambda_1}$ in (26). Otherwise, set $z = \sqrt{\lambda_1/\mu_1}$ and use the known value of $g(1,0)$.

(ii) $\sqrt{\mu_1/\lambda_1} > \alpha_3$.

In this case, $g(x,0)$ has a simple pole at α_3, as does $g_{\{1\}}(x)$. However, $g(x,0)$ can be continued as a meromorphic function at least up to the circle $|x| = \sqrt{\mu_1/\lambda_1}$, maintaining the validity of (21). This is done by exploiting the fact that when x is outside the unit disc, a branch $y(x)$ is inside the unit disc; there $g(0,y)$ and $g_{\{2\}}(y)$ are analytic. The residue of $g(x,0)$ at α_3 may be found by observing that the function $\mu_2(y-1)g(x,0) + \eta_2 y g_{\{1\}}(x)$ does not have a pole at that point. Thus,

$$g(x,0) = \varphi(x) + \frac{r}{x - \alpha_3} \; , \tag{27}$$

where $\varphi(x)$ is analytic,

$$r = \frac{\eta_2}{\mu_2}\frac{y(\alpha_3)}{[1 - y(\alpha_3)]} c_1 \; , \tag{28}$$

and c_1 is the appropriate constant in (18).

In place of the Dirichlet problem (23) we now have one for $\varphi(x)$:

$$Im[(\frac{\mu_1}{\lambda_1} - x)\varphi(x)] = u(x) \quad , \quad |x| = \sqrt{\frac{\mu_1}{\lambda_1}} \; , \tag{29}$$

where

$$u(x) = \frac{y(x)}{\mu_2[1 - y(x)]} Im\{(\frac{\mu_1}{\lambda_1} - x)[\eta_2 g_{\{1\}}(x) + \eta_1 g_{\{2\}}(y(x)) + \frac{r}{x - \alpha_3}]\} \; .$$

An expression similar to (26) can be obtained for $\varphi(z)$ in the disc $|z| < \sqrt{\mu_1/\lambda_1}$, with the unknown constant determined from the known value of $g(1,0)$.

In the event that $\alpha_3 = \sqrt{\mu_1/\lambda_1}$, the singularity of $u(x)$ at $x = \alpha_3$ can be removed by multiplying both sides of (29) by $(x - \alpha_3)(\overline{x} - \alpha_3)$, which is real.

The other unknown function, $g(0,y)$, can be obtained in a similar fashion, but also directly from (21). Having $g(x,0)$ and $g(0,y)$, $g(x,y)$ is determined by (20).

6 CONCLUSIONS

We have made some progress with a rather difficult problem. The ability to determine the marginal distributions of queue sizes is clearly useful. The optimal splitting of the input stream among the gateways is not easy to guess, even in the case of $N = 2$. More experimentation in that direction is required. The joint distribution of two queues could also provide relevant information: for instance, the asymptotic distribution of one queue size, given that the other one is very large.

A related, but radically different model is obtained by assuming that a server breakdown leaves the currently queued jobs (and perhaps the one being served) in place during the subsequent repair period. New arrivals are redirected as before. That analysis appears to be a more difficult problem, which would be a worthy object of further study.

REFERENCES

1. B. Avi-Itzhak and P. Naor, *Some Queueing Problems with the Service Station Subject to Breakdowns*, Operations Research, 11, pp. 303-320, 1963.
2. G. Fayolle and R. Iasnogorodski, *Two Coupled Processors: The reduction to a Riemann-Hilbert Problem*, Z. Wahrscheinlichkeitstheorie, 47, pp. 325-351, 1979.
3. G. Fayolle, P.J.B. King and I. Mitrani, *The Solution of Certain Two-Dimensional Markov Models*, J. Appl. Prob., 19, pp. 295-308, 1982.
4. W. Feller, *An Introduction to Probability Theory and its Applications, Vol. 1*, Wiley, 1968.
5. F.D. Gakhov, *Boundary Value Problems*, Addison Wesley, 1966.
6. K. Hoffman and R. Kunze, *Linear Algebra*, Prentice Hall, 1971.
7. L. Kleinrock, *Queueing Systems, Vol. 1*, Wiley, 1975.
8. I.S. Gradshteyn and M. Ryzhik, *Table of Integrals, Series, and Products*, Academic Press, 1980.
9. A.G. Konheim, I. Meilijson and A. Melkman, *Processor Sharing of Two Parallel Lines*, J. Appl. Prob., 18, pp. 952-956, 1981.
10. N. Mikou, *Modelling of Queueing Networks with Breakdowns*, Doctoral Thesis, Univ. Paris-Sud, 1981.
11. I. Mitrani, *Networks of Unreliable Computers*, in *Computer Architectures and Networks* (eds. E. Gelenbe and R. Mahl), North-Holland, 1974.
12. I. Mitrani and B. Avi-Itzhak, *A Many-Server Queue with Service Interruptions*, Operations Research, 16, 3, pp.628-638, 1968.
13. N.I. Muskhelishvili, *Singular Integral Equations*, P. Noordhoff, 1953.
14. B. Sengupta, *A Queue with Service Interruptions in an Alternating Markovian Environment*, Operations Research, 38, pp. 308-318, 1990.
15. H.C. White and L.S. Christie, *Queueing with Preemptive Priorities or with Breakdown*, Operations Research, 6, pp. 79-95, 1958.

G-Networks with Multiple Class Negative and Positive Customers*

Jean-Michel Fourneau†, Erol Gelenbe‡, Rina Suros§
† *Laboratoire PRI3M, Université de Versailles*
45 Avenue des Etats-Unis, Versailles, France
‡ *Department of Electrical Engineering, Duke University*
Durham, NC 27709-0291, USA
EHEI, Université René Descartes
45 Rue des Saints Pères, Paris, France
§ *Escuela de Computacion, Universidad Central de Venezuela*
Caracas, Venezuela

A new class of queueing networks with "negative and positive" customers was introduced and shown to have a non-standard product form [4]; since then this model has undergone several generalizations [5, 6, 7, 8]. Positive customers are identical to the usual customers of a queueing network, while a negative customer which arrives to a queue simply destroys a positive customer. We call these generalized queueing networks G-Networks. In this paper we extend the basic model of [4] to the case of multiple classes of positive customers, and multiple classes of negative customers. As in known multiple class queueing networks a positive customer class is characterized by the routing probabilities and the service rate parameter at each service center while negative customers of different classes may have different "customer destruction" capabilities. In the present paper all service time distributions are exponential and the service centers can be of the following types: FIFO (first-in-first-out), LIFO/PR (last-in-first-out with preemption), PS (processor sharing), with class dependent service rates.

1 INTRODUCTION

In a recent work [4], a new class of queueing networks in which customers are either "negative" or "positive" was introduced. Positive customers enter a queue and receive service as ordinary queueing network customers. A negative customer will vanish if it arrives to an empty queue, and it will reduce by one the number of positive customers in queue otherwise. Negative customers do not receive service. Positive customers which leave a queue to enter another queue can become negative or remain positive.

---

*Research supported by the European Grant BRA-QMIPS of CEC DG XIII, and by Programme C3-CNRS

It has been shown [4] that networks of queues with a single class of positive and negative customers have a product form solution if the external positive or negative customer arrivals are Poisson, the service times of positive customers are exponential and independent, and if the movement of customers between queues is Markovian.

The single server queue with negative and positive customers has been examined in [5]. Stability conditions for these networks have been discussed in [6], while "triggers" which are specific customers which can order the re-routing of customers [7], and batch removal of customers by negative customers, have been introduced in [8, 9]. We call these generalized queueing networks "G-Networks" in order to distinguish them from the usual queueing network models.

G-Networks can be used to represent a variety of systems. The initial model in [4] was motivated by the analogy with neural networks [3]: each queue represents a neuron, and customers represent excitation (positive) or inhibition (negative) signals. Note that signals in biophysical neurons also take the form of random trains of impulses of constant size, much as customers traveling through a queueing network.

Another possible application is to multiple resource systems: positive customers can be considered to be resource requests, while negative customers can correspond to decisions to cancel such requests. One can also consider the analogy with the traditional "P" and "V" operations on semaphores in operating systems.

In this paper we extend the model to G-Networks with **multiple classes of positive customers and one or more classes of negative customers**. In particular, we consider three types of service centers with their corresponding service disciplines:

- Type 1 : first-in-first-out (FIFO),
- Type 2 : processor sharing (PS),
- Type 4 : last-in-first-out with preemptive resume priority (LIFO/PR).

Here we are using the usual terminology related to the BCMP theorem [1], exclude from our model the Type 3 service centers with an infinite number of servers **since they will not be covered by our results**.

In Section 3 we will prove that these multiple class G-Networks, with Type 1, 2 and 4 service centers, have product form. Due to the non-linearity of the traffic equations for these models [4] the existence and uniqueness of their solutions have to be addressed with some care. This issue will be examined in Section 4 with techniques similar to those developed in [6].

2 THE MODEL

We consider networks with an arbitrary number N of queues, an arbitrary number of positive customer classes K, and an arbitrary number of negative customer classes S. As in [4] we are only interested in open G-Networks.

Indeed, if the system is closed, then the total number of customers will decrease as long as there are negative customers in the network.

External arrival streams to the network are independent Poisson processes concerning positive customers of some class k or negative customers of some class c. We denote by $\Lambda_{i,k}$ the external arrival rate of *positive* customers of class k to queue i and by $\lambda_{i,m}$ the external arrival rate of *negative* customers of class m to queue i.

Only positive customers are served, and after service they may change class, service center and nature (positive to negative), or depart from the system. The movement of customers between queues, classes and nature (positive to negative) is represented by a Markov chain.

At its arrival in a non empty queue, a negative customer selects a positive customer in the queue in accordance with the service discipline at this station. If the queue is empty , then the negative customer simply disappears. Once the target is selected, the negative customer tries to destroy the selected customer. A negative customer of some class m succeeds destroying the selected positive customer of some class k at service center i with probability $K_{i,m,k}$. And with probability $(1 - K_{i,m,k})$ it does not succeed. A negative customer disappears as soon as it tries to destroy its targeted customer. Recall that a negative customer is either exogenous, or is obtained by the transformation of a positive customer as it leaves a queue.

A positive customer of class k which leaves queue i (after finishing service) goes to queue j as a positive customer of class l with probability $P^+[i,j]\ [k,l]$, or as a negative customer of class m with probability $P^-[i,j]\ [k,m]$. It may also depart from the network with probability $d[i,k]$.

We shall assume that $P^+[i,i][k,l] = 0$ and $P^-[i,i][k,m] = 0$ for all i,k,l,m so that customers are not allowed to return to the queue they have just left. Though this assumption is without loss of generality for positive customers, it is indeed a restriction of the model for negative customers since it avoids double simultaneous customer "departures" at a queue.

Obviously we have for all i,k

$$\sum_{j=1}^{N}\sum_{l=1}^{R} P^+[i,j][k,l] \; + \; \sum_{j=1}^{N}\sum_{m=1}^{S} P^-[i,j][k,m] \; + \; d[i,k] = 1 \tag{1}$$

We assume that all service centers have exponential service time distributions. In the three types of service centers, each class of positive customers may have a distinct service rate $\mu_{i,k}$.

However, when the service center is of Type 1 (FIFO) the following quantity, which takes into account the service rate, must be a constant for each class k of positive customer.

$$\mu_{i,k} + \sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} = c_i \tag{2}$$

The following constraints on the deletion discipline and deletion probability are assumed to exist.

- For a Type 1 server, an arriving negative customer selects the customer being served; and the following constraint must hold for all stations i of type 1 and classes of negative customers m such that $\sum_{j=1}^{N}\sum_{l=1}^{R} P^{-}[j,i][l,m] > 0$

  $$\text{for all classes of positive customers k and p}, K_{i,m,k} = K_{i,m,p} \qquad (3)$$

  This constraint implies that a negative customer of some class m arriving from the network does not "distinguish" between the positive customer classes it will try to delete, and that it will treat them all in the same manner.

- For a Type 2 server, the probability that any one positive customer of the queue is selected by the arriving negative customer is $1/c$ if c is the total number of customers in queue.

- In a Type 4 server, the only restriction is that the positive customer selected by a negative customer is the one in service.

For Type 1 service centers, one may consider the following conditions which are simpler than (2) and (3):

$$\begin{aligned} \mu_{ik} &= \mu_{ip} \, , \\ K_{i,m,k} &= K_{i,m,p} \end{aligned} \qquad (4)$$

for all classes of positive customers k and p, and all classes of negative customers m. Note however that these new conditions are more restrictive, though they do imply that (2, 3) hold.

2.1 State representation

We shall denote the state at time t of the queueing network by the vector $x(t) = (x_1(t), ..., x_N(t))$. Here $x_i(t)$ represents the state of service center i. The vector $x = (x_1, ..., x_N)$ will denote a particular value of the state and $|x_i|$ will be the total number of customers in queue i for state x.

For Type 1 and Type 4 servers, the instantaneous value of the state x_i of server i is represented by the vector $(r_{i,j})$ whose length is the number of customers in the queue and whose *jth* element is the class index of the *jth* customer in the queue. Furthermore, the customers are ordered according to the service order (FIFO or LIFO); it is always the customer at the head of the list which is in service. We denote by $r_{i,1}$ the class number of the customer in service and by $r_{i,\infty}$ the class number of the last customer in the queue.

For a PS (Type 2) service station, the instantaneous value of the state x_i is represented by the vector $(x_{i,k})$ which is the number of customers of class k in queue i.

3 MAIN RESULTS

Let $\Pi(x)$ denote the stationary probability distribution of the state of the network, if it exists. The following result establishes the product form solution of the network being considered.

THEOREM 1 *Consider a G-Network with the restrictions indicated above. If the system of non-linear equations*

$$
\begin{aligned}
q_{i,k} &= \frac{\Lambda_{i,k} + \Lambda^{+}_{i,k}}{\mu_{i,k} + \sum_{m=1}^{S} K_{i,m,k}[\lambda_{i,m} + \lambda^{-}_{i,m}]} && (5)\\
\Lambda^{+}_{i,k} &= \sum_{j=1}^{N}\sum_{l=1}^{R} P^{+}[j,i][l,k]\mu_{j,l}q_{j,l} && (6)\\
\lambda^{-}_{i,m} &= \sum_{j=1}^{N}\sum_{l=1}^{R} P^{-}[j,i][l,m]\mu_{j,l}q_{j,l} && (7)
\end{aligned}
$$

has a solution such that

$$\textit{for each pair } i,k: \quad 0 < q_{i,k} \quad \textit{and} \quad \textit{for each station } i: \quad \sum_{k=1}^{R} q_{i,k} < 1$$

then the stationary distribution of the network state is

$$\Pi(x) = G\prod_{i=1}^{N} g_i(x_i) \tag{8}$$

where each $g_i(x_i)$ depends on the type of service center i. The $g_i(x_i)$ in (8) have the following form :

FIFO *If the service center is of Type 1, then*

$$g_i(x_i) = \prod_{n=1}^{|x_i|}[q_{i,r_{i,n}}] \tag{9}$$

PS *If the service center is of Type 2, then*

$$g_i(x_i) = |x_i|!\prod_{k=1}^{R}\frac{(q_{i,k})^{x_{i,k}}}{x_{i,k}!} \tag{10}$$

LIFO/PR *If the service center is of Type 4, then*

$$g_i(x_i) = \prod_{n=1}^{|x_i|}[q_{i,r_{i,n}}] \tag{11}$$

and G is the normalization constant.

The proof is based on simple algebraic manipulations of global balance equations, since it is not possible to use the "local balance" equations for customer classes at stations because of the effect of negative customer arrivals. We begin with some technical lemmas.

LEMMA 1 *The following flow equation is satisfied:*

$$\sum_{i=1}^{N}\sum_{k=1}^{R} q_{i,k}\mu_{i,k}(1-d[i,k]) = \sum_{i=1}^{N}\sum_{k=1}^{R}\Lambda_{i,k}^{+} + \sum_{i=1}^{N}\sum_{m=1}^{S}\lambda_{i,m}^{-} \tag{12}$$

Proof : Consider (6), then sum it for all the stations and all the classes and exchange the order of summations in the right-hand side of the equation :

$$\sum_{i=1}^{N}\sum_{k=1}^{R}\Lambda_{i,k}^{+} = \sum_{j=1}^{N}\sum_{l=1}^{R}\mu_{j,l}q_{j,l}(\sum_{i=1}^{N}\sum_{k=1}^{R}P^{+}[j,i][l,k])$$

Similarly, using equation (7)

$$\sum_{i=1}^{N}\sum_{m=1}^{S}\lambda_{i,m}^{-} = \sum_{j=1}^{N}\sum_{l=1}^{R}\mu_{j,l}q_{j,l}(\sum_{i=1}^{N}\sum_{m=1}^{S}P^{-}[j,i][l,m]).$$

And,

$$\sum_{i=1}^{N}\sum_{k=1}^{R}\Lambda_{i,k}^{+} + \sum_{i=1}^{N}\sum_{m=1}^{S}\lambda_{i,m}^{-} = \sum_{j=1}^{N}\sum_{l=1}^{R}\mu_{j,l}q_{j,l}(\sum_{i=1}^{N}\sum_{k=1}^{R}P^{+}[j,i][l,k]$$

$$+\sum_{i=1}^{N}\sum_{m=1}^{S}P^{-}[j,i][l,m]).$$

According to the definition of the routing matrix P (equation 1), we have

$$\sum_{i=1}^{N}\sum_{k=1}^{R}\Lambda_{i,k}^{+} + \sum_{i=1}^{N}\sum_{m=1}^{S}\lambda_{i,m}^{-} = \sum_{j=1}^{N}\sum_{l=1}^{R}\mu_{j,l}q_{j,l}(1-d[j,l]).$$

Thus the proof of the lemma is complete.

□

In order to carry out algebraic manipulations of the stationary Chapman-Kolmogorov (global balance) equations, we introduce some notation and develop intermediate results:

- The state dependent service rates for customers at service center j will be denoted by $M_{j,l}(x_j)$ where x_j refers to the state of the service center and l is the class of the customer concerned. From the definition of the service rate $\mu_{j,l}$, we obtain for the three types of stations :

  FIFO and LIFO/PR $M_{j,l}(x_j) = \mu_{j,l} 1_{\{r_{j,1}=l\}}$,
  PS $M_{j,l}(x_j) = \mu_{j,l} \frac{x_{j,l}}{|x_j|}$.

- $N_{j,l}(x_j)$ is the deletion rate of class l positive customers due to external arrivals of all the classes of negative customers

  FIFO and LIFO/PR $N_{j,l}(x_j) = 1_{\{r_{j,1}=l\}} \sum_{m=1}^{S} K_{j,m,l}\lambda_{j,m}$
  PS $N_{j,l}(x_j) = \frac{x_{j,l}}{|x_j|} \sum_{m=1}^{S} K_{j,m,l}\lambda_{j,m}$.

- $A_{j,l}(x_j)$ is the condition which establishes that it is possible to reach state x_j by an arrival of a positive customer of class l

  FIFO $A_{j,l}(x_j) = 1_{\{r_{j,\infty}=l\}}$,
  LIFO/PR $A_{j,l}(x_j) = 1_{\{r_{j,1}=l\}}$,
  PS $A_{j,l}(x_j) = 1_{\{|x_{j,l}|>0\}}$.

- $Z_{j,l,m}(x_j)$ is the probability that a negative customer of class m, arriving from the network, will delete a positive customer of class l.

  FIFO and LIFO/PR $Z_{j,l,m}(x_j) = 1_{\{r_{j,1}=l\}} K_{j,m,l}$
  PS $Z_{j,l,m}(x_j) = \frac{x_{j,l}}{|x_j|} K_{j,m,l}$.

- $Y_{j,m}(x_j)$ is the probability that a negative customer of class m which enters a non empty queue, will not delete a positive customer.

  FIFO and LIFO/PR $Y_{j,m}(x_j) = \sum_{l=1}^{R} 1_{\{r_{j,1}=l\}}(1 - K_{j,m,l})$
  PS $Y_{j,m}(x_j) = \sum_{l=1}^{R}(1 - K_{j,m,l})\frac{x_{j,l}}{|x_j|}$.

Denote by $(x_j + e_{j,l})$ the state of station j obtained by **adding to the server a positive customer of class** l. Denote by $(x_i - e_{i,k})$ the state obtained by removing from the end of the list a class k customer (if it exists, since otherwise $(x_i - e_{i,k})$ will not be defined).

LEMMA 2 *For any Type 1, 2, or 4 service center, the following relations hold:*

$$M_{j,l}(x_j + e_{j,l})\frac{g_j(x_j + e_{j,l})}{g_j(x_j)} = \mu_{j,l} q_{j,l} \tag{13}$$

$$N_{j,l}(x_j + e_{j,l})\frac{g_j(x_j + e_{j,l})}{g_j(x_j)} = \sum_{m=1}^{S} K_{j,m,l} q_{j,l} \lambda_{j,m} \tag{14}$$

$$Z_{j,l,m}(x_j + e_{j,l})\frac{g_j(x_j + e_{j,l})}{g_j(x_j)} = K_{j,m,l} q_{j,l} \tag{15}$$

The proof is purely algebraic.
Remark : As a consequence, we have from equations (6), (7) and (13):

$$\Lambda^+_{i,k} = \sum_{j=1}^{N}\sum_{l=1}^{R} M_{j,l}(x_j + e_{j,l})\frac{g_j(x_j + e_{j,l})}{g_j(x_j)}P^+[j,i][l,k] \tag{16}$$

and

$$\lambda^-_{i,m} = \sum_{j=1}^{N}\sum_{l=1}^{R} M_{j,l}(x_j + e_{j,l})\frac{g_j(x_j + e_{j,l})}{g_j(x_j)}P^-[j,i][l,m] \tag{17}$$

LEMMA 3 *Let i be any Type 1, 2, or 4 station, and let $\Delta_i(x_i)$ be:*

$$\begin{aligned}\Delta_i(x_i) &= \sum_{m=1}^{S} \lambda^-_{i,m} Y_{i,m}(x_i)\\ &- \sum_{k=1}^{R}(M_{i,k}(x_i) + N_{i,k}(x_i))\\ &+ \sum_{k=1}^{R} A_{i,k}(x_i)(\Lambda_{i,k} + \Lambda^+_{i,k})\frac{g_i(x_i - e_{i,k})}{g_i(x_i)}\end{aligned}$$

Then for the three types of service centers, $1_{\{|x_i|>0\}}\Delta_i(x_i) = \sum_{m=1}^{S} \lambda^-_{i,m} 1_{\{|x_i|>0\}}$.

The proof of Lemma 3 is in the appendix. □

Let us now turn to the proof of the Theorem 1. Consider the global balance equation for the considered networks:

$$\begin{aligned}\Pi(x)[\sum_{j=1}^{N}\sum_{l=1}^{R}(\Lambda_{j,l} + M_{j,l}(x_j)1_{\{|x_j|>0\}} + N_{j,l}(x_j)1_{\{|x_j|>0\}})]\\ = \sum_{j=1}^{N}\sum_{l=1}^{R}\Pi(x - e_{j,l})\Lambda_{j,l}A_{j,l}(x_j)1_{\{|x_j|>0\}}\\ + \sum_{j=1}^{N}\sum_{l=1}^{R}\Pi(x + e_{j,l})N_{j,l}(x_j + e_{j,l})\\ + \sum_{j=1}^{N}\sum_{l=1}^{R}\Pi(x + e_{j,l})M_{j,l}(x_j + e_{j,l})d[j,l]\\ + \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{R}\sum_{l=1}^{R} M_{j,l}(x_j + e_{j,l})\Pi(x - e_{i,k} + e_{j,l})\end{aligned}$$

$$
\begin{aligned}
& \qquad\qquad P^{+}[j,i][l,k]A_{i,k}(x_i)1_{\{|x_i|>0\}} \\
+ & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{R}\sum_{l=1}^{R}\sum_{m=1}^{S} M_{j,l}(x_j+e_{j,l})\Pi(x+e_{i,k}+e_{j,l}) \\
& \qquad\qquad P^{-}[j,i][l,m]Z_{i,k,m}(x_i+e_{i,k}) \\
+ & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{l=1}^{R}\sum_{m=1}^{S} M_{j,l}(x_j+e_{j,l})\Pi(x+e_{j,l}) \\
& \qquad\qquad P^{-}[j,i][l,m]Y_{i,m}(x_i)1_{\{|x_i|>0\}} \\
+ & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{l=1}^{R}\sum_{m=1}^{S} M_{j,l}(x_j+e_{j,l})\Pi(x+e_{j,l})P^{-}[j,i][l,m]1_{\{|x_i|=0\}}
\end{aligned}
$$

We divide both sides by $\Pi(x)$ and we assume that there is a product form solution. Then, we apply lemma 2.

$$
\begin{aligned}
& \sum_{j=1}^{N}\sum_{l=1}^{R}(\Lambda_{j,l}+M_{j,l}(x_j)1_{\{|x_j|>0\}}+N_{j,l}(x_j)1_{\{|x_j|>0\}}) \\
= & \sum_{j=1}^{N}\sum_{l=1}^{R}\frac{g_j(x_j-e_{j,l})}{g_j(x_j)}\Lambda_{j,l}A_{j,l}(x_j)1_{\{|x_j|>0\}} \\
+ & \sum_{j=1}^{N}\sum_{l=1}^{R}\sum_{m=1}^{S}\lambda_{j,m}K_{j,m,l}q_{j,l} \; + \; \sum_{j=1}^{N}\sum_{l=1}^{R}\mu_{j,l}q_{j,l}d[j,l] \\
+ & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{R}\sum_{l=1}^{R}\mu_{j,l}q_{j,l}P^{+}[j,i][l,k]A_{i,k}(x_i)\frac{g_i(x_i-e_{i,k})}{g_i(x_i)}1_{\{|x_i|>0\}} \\
+ & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{R}\sum_{l=1}^{R}\sum_{m=1}^{S}\mu_{j,l}q_{j,l}P^{-}[j,i][l,m]K_{i,m,k}q_{i,k} \\
+ & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{l=1}^{R}\sum_{m=1}^{S}\mu_{j,l}q_{j,l}P^{-}[j,i][l,m]Y_{i,m}(x_i)1_{\{|x_i|>0\}} \\
+ & \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{l=1}^{R}\sum_{m=1}^{S}\mu_{j,l}q_{j,l}P^{-}[j,i][l,m]1_{\{|x_i|=0\}}
\end{aligned}
$$

After some substitution, we group the first and the fourth terms of the right side of the equation.

$$
\sum_{j=1}^{N}\sum_{l=1}^{R}(\Lambda_{j,l}+M_{j,l}(x_j)1_{\{|x_j|>0\}}+N_{j,l}(x_j)1_{\{|x_j|>0\}})
$$

$$\begin{aligned}
&= \sum_{j=1}^{N}\sum_{l=1}^{R} 1_{\{|x_j|>0\}} \frac{g_j(x_j - e_{j,l})}{g_j(x_j)} A_{j,l}(x_j)(\Lambda_{j,l} + \Lambda^{+}_{j,l}) \\
&+ \sum_{j=1}^{N}\sum_{l=1}^{R}\sum_{m=1}^{S} \lambda_{j,m} K_{j,m,l} q_{j,l} \\
&+ \sum_{j=1}^{N}\sum_{l=1}^{R} \mu_{j,l} q_{j,l} d[j,l] \\
&+ \sum_{i=1}^{N}\sum_{k=1}^{R}\sum_{m=1}^{S} \lambda^{-}_{i,m} K_{i,m,k} q_{i,k} \\
&+ \sum_{i=1}^{N}\sum_{m=1}^{S} \lambda^{-}_{i,m} Y_{i,m}(x_i) 1_{\{|x_i|>0\}} \\
&+ \sum_{i=1}^{N}\sum_{m=1}^{S} \lambda^{-}_{i,m} 1_{\{|x_i|=0\}}
\end{aligned}$$

We add to both sides the quantity $\sum_{j=1}^{N}\sum_{l=1}^{R} \mu_{j,l} q_{j,l}(1 - d[j,l])$ and factorize three terms in the right side

$$\begin{aligned}
&\sum_{j=1}^{N}\sum_{l=1}^{R} (\Lambda_{j,l} + M_{j,l}(x_j) 1_{\{|x_j|>0\}} + N_{j,l}(x_j) 1_{\{|x_j|>0\}}) \\
&\quad + \sum_{j}\sum_{l} \mu_{j,l} q_{j,l}(1 - d[j,l]) \\
&\quad = \sum_{j=1}^{N}\sum_{l=1}^{R} 1_{\{|x_j|>0\}} \frac{g_j(x_j - e_{j,l})}{g_j(x_j)} A_{j,l}(x_j)(\Lambda_{j,l} + \Lambda^{+}_{j,l}) \\
&\quad + \sum_{j=1}^{N}\sum_{l=1}^{R} q_{j,l}(\mu_{j,l} + \sum_{m=1}^{S} \lambda_{j,m} K_{j,m,l} + \sum_{m=1}^{S} \lambda^{-}_{j,m} K_{j,m,l}) \\
&\quad + \sum_{i=1}^{N}\sum_{m=1}^{S} \lambda^{-}_{i,m} Y_{i,m}(x_i) 1_{\{|x_i|>0\}} \\
&\quad + \sum_{i=1}^{N}\sum_{m=1}^{S} \lambda^{-}_{i,m} 1_{\{|x_i|=0\}}
\end{aligned}$$

We substitute the value of $q_{i,k}$ in the second term. Then, we cancel the term $\Lambda_{j,l}$ which appears on both sides and we group terms to obtain :

$$\sum_{j=1}^{N}\sum_{l=1}^{R} \mu_{j,l} q_{j,l}(1 - d[j,l]) = \sum_{j=1}^{N}\sum_{l=1}^{R} \Lambda^{+}_{j,l} + \sum_{i=1}^{N} 1_{\{|x_i|>0\}} \Delta_i(x_i) \qquad (18)$$

$$+\sum_{i=1}^{N}\sum_{m=1}^{S} \lambda^-_{i,m} 1_{\{|x_i|=0\}}$$

where $\Delta_i(x_i)$ is defined in Lemma 3.

In Lemma 3, we have shown that $1_{\{|x_i|>0\}}\Delta_i(x_i)$ is equal to $\sum_{m=1}^{S} \lambda^-_{i,m} 1_{\{|x_i|>0\}}$ for the three types of service centers. Thus,

$$\sum_{j=1}^{N}\sum_{l=1}^{R} \mu_{j,l} q_{j,l}(1 - d[j,l]) =$$
$$\sum_{j=1}^{N}\sum_{l=1}^{R} \Lambda^+_{j,l} + \sum_{i=1}^{N}\sum_{m=1}^{S} \lambda^-_{i,m}(1_{\{|x_i|=0\}} + 1_{\{|x_i|>0\}})$$

Finally, Lemma 1 shows that this flow equation is satisfied. This concludes the proof. □

As in the BCMP [1] theorem, we can also compute the steady state distribution of the number of customers of each class in each queue. Let y_i be the vector whose elements are $(y_{i,k})$ the number of customers of class k in station i. Let y be the vector of vectors (y_i).

THEOREM 2 *If the system of equations (5), (6) and (7) has a solution then, the steady state distribution* $\pi(y)$ *is given by*

$$\pi(y) = \prod_{i=1}^{N} h_i(y_i) \tag{19}$$

where the marginal probabilities $h_i(y_i)$ *have the following form :*

$$h_i(y_i) = (1 - \sum_{k=1}^{R} q_{i,k})|y_i|! \prod_{k=1}^{R} [(q_{i,k})^{y_{i,k}}/y_{i,k}!] \tag{20}$$

4 EXISTENCE OF THE SOLUTION TO THE TRAFFIC EQUATIONS

Unlike BCMP or Jackson networks [1], the customer flow equations (5), (6) and (7) of the model we consider are non-linear. Therefore issues of existence and uniqueness of their solutions have to be examined.

In particular, our key result depends on the existence of solutions to (5), (6), (7). Thus the issue of existence and uniqueness of solutions to these traffic equations is central to our work.

Define the following matrices:

Λ^+ with elements $\Lambda^+_{i,k}$

λ^- with elements $\lambda^-_{i,k}$

Λ with elements $\Lambda_{i,k}$, and

λ with elements $\lambda_{i,k}$

Furthermore, denote by P^+ the matrix with elements $\{P^+[i,j]\}$, and by P^- the matrix whose elements are $\{P^-[i,j]\}$.

Let F be a diagonal matrix with elements $0 \leq F_{i,k} \leq 1$. Equations (6) and (7) inspire us to write the following equation:

$$\Lambda^+ = \Lambda^+ FP^+ + \Lambda, \qquad \lambda^- = \Lambda^+ FP^- + \lambda \tag{21}$$

or, denoting the identity matrix by $\mathbf{I}$, as

$$\begin{aligned} \Lambda^+(\mathbf{I} - FP^+) &= \Lambda, & (22)\\ \lambda^- &= \Lambda^+ FP^- + \lambda. & (23) \end{aligned}$$

Proposition 1: If P^+ is a substochastic matrix which does not contain ergodic classes, then equations (22) and (23) have a solution (Λ^+, λ^-).

PROOF: The series $\sum_{n=0}^{\infty}(FP^+)^n$ is geometrically convergent, since $F \leq \mathbf{I}$, and because – by assumption – P^+ is substochastic and does not contain any ergodic classes (see Kemeny and Snell [6] pp 43 ff]). Therefore we can write (22) as

$$\Lambda^+ = \Lambda \sum_{n=0}^{\infty}(FP^+)^n, \tag{24}$$

so that (23) becomes

$$\lambda^- - \lambda = \Lambda \sum_{n=0}^{\infty}(FP^+)^n FP^-. \tag{25}$$

Now denote $z = \lambda^- - \lambda$, and call the vector function

$$G(z) = \Lambda \sum_{n=0}^{\infty}(FP^+)^n FP^-$$

Note that the dependency of G on z comes from F, which depends on λ^-.

It can be seen that $G : [\mathbf{0}, G(0)] \rightarrow [\mathbf{0}, G(0)]$ and that it is continuous. Therefore, by Brouwer's fixed-point thorem

$$z = G(z) \tag{26}$$

has a fixed point z^*. This fixed point will yield the solution of (22) and (23) as:

$$\lambda^-(\mathrm{y}^*) = \lambda + \mathrm{y}^*, \qquad \Lambda^+(\mathrm{z}^*) = \Lambda \sum_{n=0}^{\infty}(F(\mathrm{z}^*)P^+)^n, \tag{27}$$

completing the proof of Proposition 1.

Proposition 2 Equations (6), (7) have a solution.

PROOF: This result is a direct consequence of Proposition 1, since we can see that (5), (6) and (7) are a special instance of (21). Indeed, it suffices to set

$$F_{i,k} = \frac{\mu_{i,k}}{\mu_{i,k} + \sum_{m=1}^{S} K_{i,m,k}[\lambda_{i,m} + \lambda_{i,m}^{-}]} \tag{28}$$

and to notice that $0 \leq F_{i,k} \leq 1$, and that (6), (7) now have taken the form of the generalized traffic equations (21). This completes the proof of Proposition 2.

The above two propositions state that the traffic equations *always* have a solution. Of course, the product form (8) will only exist if the resulting network is stable. The stability condition is summarized below.

THEOREM 3 *Let z^* be a solution of $z = G(z)$ obtained by setting F as in (27). Let $\lambda^-(z^*)$, $\Lambda^+(z^*)$ be the corresponding traffic values, and let the $q_{i,k}(z^*)$ be obtained from (5) as a consequence. Then the G-network is stable if all of the $0 \leq q_{i,k}(z^*) < 1$ for all i, k. Otherwise it is unstable.*

5 CONCLUSIONS

In this paper we have considered networks of queues with multiple classes of positive customers and multiple classes of negative customers. We have shown that these new networks have product form when all service centers – with the exception of the "infinite server" case – are similar to the service centers considered in the BCMP theorem [1] with class dependent exponential service time distributions. As in the BCMP theorem, FIFO service centers have a single class of exponential service times.

Further extensions of these results to more complex service distributions or more complex interactions between positive and negative customers are currently being considered.

REFERENCES

1. Baskett F., Chandy K., Muntz R.R., Palacios F.G., Open, closed and mixed networks of queues with different classes of customers, *Journal ACM* 22: 248–260, 1975.
2. Fourneau J.M., Computing the steady-state distribution of networks with positive and negative customers, LRI Report, 13th IMACS World Congress on Computation and Applied Mathematics, Dublin, 1991.
3. Gelenbe E., Random neural Networks with Negative and Positive Signals and Product Form Solution, *Neural Computation* 1: 502–510, 1990.

4. Gelenbe E., Product form queueing networks with negative and positive customers, *Journal of Applied Probability* 28: 656–663, 1991.
5. Gelenbe E., Glynn P., Sigmann K. Queues with negative customers, *Journal of Applied Probability* 28: 245–250, 1991.
6. Gelenbe E., Schassberger R., Stability of G-Networks, *Probability in the Engineering and Informational Sciences* 6: 271–276, 1992.
7. Gelenbe E., G-Networks with triggered customer movement, *Journal of Applied Probability* 30: 742–748, 1993.
8. Gelenbe E., G-Networks with signals and batch removal, *Probability in the Engineering and Informational Sciences* 7: 335–342, 1993.
9. Henderson, W., Northcote, B.S., Taylor, P.G., Geometric equilibrium distributions for queues with interactive batch departures, *Annals of Operations Research* 48: 493–511, 1994.

6 APPENDIX

Proof of Lemma 3 : The proof consists in algebraic manipulations for the three types of stations.

LIFO/PR First consider an arbitrary LIFO station and remember the definition of Δ_i :

$$\begin{aligned}
1_{\{|x_i|>0\}}\Delta_i(x_i) &= 1_{\{|x_i|>0\}}\sum_{k=1}^{R} A_{i,k}(x_i)(\Lambda_{i,k}+\Lambda^{+}_{i,k})\frac{g_i(x_i-e_{i,k})}{g_i(x_i)} \\
&- 1_{\{|x_i|>0\}}\sum_{k=1}^{R} M_{i,k}(x_i) \;-\; 1_{\{|x_i|>0\}}\sum_{k=1}^{R} N_{i,k}(x_i) \\
&+ 1_{\{|x_i|>0\}}\sum_{m=1}^{S} \lambda^{-}_{i,m} Y_{i,m}(x_i)
\end{aligned}$$

Then, we substitute the values of $Y_{i,m}$, $M_{i,k}$, $N_{i,k}$ and $A_{i,k}$ for a LIFO station:

$$\begin{aligned}
1_{\{|x_i|>0\}}\Delta_i(x_i) &= 1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}\ (\Lambda_{i,k}+\Lambda^{+}_{i,k})/q_{i,k} \\
&- 1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}\ \mu_{i,k} \\
&- 1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}\ \sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} \\
&+ 1_{\{|x_i|>0\}}\sum_{m=1}^{S} \lambda^{-}_{i,m}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}(1-K_{i,m,k})
\end{aligned}$$

We use the value of $q_{i,k}$ from equation 5 to obtain after some cancellations of terms:

$$\begin{aligned}
1_{\{|x_i|>0\}}\Delta_i(x_i) &= 1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}(\sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m}^{-} \\
&+ \sum_{m=1}^{S}\lambda_{i,m}^{-}(1-K_{i,m,k}) \\
&= 1_{\{|x_i|>0\}}\sum_{m=1}^{S}\lambda_{i,m}^{-}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}
\end{aligned}$$

and as $1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}} = 1_{\{|x_i|>0\}}$, we finally get the result:

$$1_{\{|x_i|>0\}}\Delta_i(x_i) = 1_{\{|x_i|>0\}}\sum_{m=1}^{S}\lambda_{i,m}^{-} \tag{29}$$

FIFO Consider now an arbitrary FIFO station:

$$\begin{aligned}
1_{\{|x_i|>0\}}\Delta_i(x_i) &= 1_{\{|x_i|>0\}}\sum_{k=1}^{R} A_{i,k}(x_i)(\Lambda_{i,k}+\Lambda_{i,k}^{+})\frac{g_i(x_i-e_{i,k})}{g_i(x_i)} \\
&- 1_{\{|x_i|>0\}}\sum_{k=1}^{R} M_{i,k}(x_i) \;-\; \sum_{k=1}^{R} 1_{\{|x_i|>0\}}N_{i,k}(x_i) \\
&+ 1_{\{|x_i|>0\}}\sum_{m=1}^{S}\lambda_{i,m}^{-}Y_{i,m}(x_i)
\end{aligned}$$

Similarly, we substitute the values of $Y_{i,m}$, $M_{i,k}$, $N_{i,k}$, $A_{i,k}$ and $q_{i,k}$:

$$\begin{aligned}
1_{\{|x_i|>0\}}\Delta_i(x_i) &= 1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,\infty}=k\}}\;(\mu_{i,k}+\sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} \\
&+ \sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m}^{-}) - 1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}\;\mu_{i,k} \\
&- \;\; 1_{\{|x_i|>0\}}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}\sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} \\
&+ 1_{\{|x_i|>0\}}\sum_{m=1}^{S}\lambda_{i,m}^{-}\sum_{k=1}^{R} 1_{\{r_{i,1}=k\}}(1-K_{i,m,k})
\end{aligned}$$

We separate the last term into two parts, and regroup terms:

$$\begin{aligned} 1_{\{|x_i|>0\}}\Delta_i(x_i) &= 1_{\{|x_i|>0\}} \sum_{k=1}^{R} 1_{\{r_{i,\infty}=k\}} \; (\mu_{i,k} + \sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} \\ &+ \sum_{m=1}^{S} K_{i,m,k}\lambda^-_{i,m}) - 1_{\{|x_i|>0\}} \sum_{k=1}^{R} 1_{\{r_{i,1}=k\}} \; (\mu_{i,k} \\ &+ \sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} + \sum_{m=1}^{S} K_{i,m,k}\lambda^-_{i,m}) \\ &+ 1_{\{|x_i|>0\}} \sum_{m=1}^{S} \lambda^-_{i,m} \sum_{k=1}^{R} 1_{\{r_{i,1}=k\}} \end{aligned}$$

But the assumptions of theorem 1 imply that we have the following relation:

$$\begin{aligned} &\sum_{k=1}^{R} 1_{\{r_{i,\infty}=k\}} \; (\mu_{i,k} + \sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} + \sum_{m=1}^{S} K_{i,m,k}\lambda^-_{i,m}) = \\ &\quad \sum_{k=1}^{R} 1_{\{r_{i,1}=k\}} \; (\mu_{i,k} + \sum_{m=1}^{S} K_{i,m,k}\lambda_{i,m} + \sum_{m=1}^{S} K_{i,m,k}\lambda^-_{i,m}) \end{aligned}$$

Thus, as $1_{\{|x_i|>0\}} \sum_{k=1}^{R} 1_{\{r_{i,1}=k\}} = 1_{\{|x_i|>0\}}$, we finally get the expected result:

$$1_{\{|x_i|>0\}}\Delta_i(x_i) = 1_{\{|x_i|>0\}} \sum_{m=1}^{S} \lambda^-_{i,m} \tag{30}$$

PS Consider now an arbitrary PS station:

$$\begin{aligned} 1_{\{|x_i|>0\}}\Delta_i(x_i) &= 1_{\{|x_i|>0\}} \sum_{k=1}^{R} A_{i,k}(x_i)(\Lambda_{i,k} + \Lambda^+_{i,k}) \frac{g_i(x_i - e_{i,k})}{g_i(x_i)} \\ &- 1_{\{|x_i|>0\}} \sum_{k=1}^{R} M_{i,k}(x_i) \;-\; \sum_{k=1}^{R} 1_{\{|x_i|>0\}} N_{i,k}(x_i) \\ &+ 1_{\{|x_i|>0\}} \sum_{m=1}^{S} \lambda^-_{i,m} Y_{i,m}(x_i) \end{aligned}$$

As usual, we substitute the values of $Y_{i,m}$, $M_{i,k}$, $N_{i,k}$, $A_{i,k}$:

$$1_{\{|x_i|>0\}}\Delta_i(x_i) = 1_{\{|x_i|>0\}} \sum_{k=1}^{R} 1_{\{|x_{i,k}|>0\}} \; \frac{(\Lambda_{i,k} + \Lambda^+_{i,k})}{q_{i,k}} \frac{x_{i,k}}{|x_i|}$$

$$
\begin{aligned}
& - \; 1_{\{|x_i|>0\}} \sum_{k=1}^{R} \mu_{i,k} \frac{x_{i,k}}{|x_i|} \\
& - \; 1_{\{|x_i|>0\}} \sum_{k=1}^{R} \frac{x_{i,k}}{|x_i|} \sum_{m=1}^{S} K_{i,m,k} \lambda_{i,m} \\
& + \; 1_{\{|x_i|>0\}} \sum_{m=1}^{S} \sum_{k=1}^{R} \lambda_{i,m}^{-} \frac{x_{i,k}}{|x_i|} (1 - K_{i,m,k})
\end{aligned}
$$

Then, we apply equation 5 to substitute $q_{i,k}$. After some cancellations of terms we obtain:

$$
\begin{aligned}
1_{\{|x_i|>0\}} \Delta_i(x_i) &= 1_{\{|x_i|>0\}} \sum_{k=1}^{R} \frac{x_{i,k}}{|x_i|} \sum_{m=1}^{S} K_{i,m,k} \lambda_{i,m}^{-} \\
&+ \; 1_{\{|x_i|>0\}} \sum_{m=1}^{S} \sum_{k=1}^{R} \lambda_{i,m}^{-} \frac{x_{i,k}}{|x_i|} (1 - K_{i,m,k})
\end{aligned}
$$

And finally,

$$
1_{\{|x_i|>0\}} \Delta_i(x_i) = 1_{\{|x_i|>0\}} \sum_{k=1}^{R} \frac{x_{i,k}}{|x_i|} \sum_{m=1}^{S} \lambda_{i,m}^{-} \tag{31}
$$

As $1_{\{|x_i|>0\}} \sum_{k=1}^{R} \frac{x_{i,k}}{|x_i|} = 1_{\{|x_i|>0\}}$, once again, we establish the relation we need. This concludes the proof of lemma 3. $\square$

Response Time Distributions in Tandem G-Networks†

Peter G. Harrison [1] Edwige Pitel [2]
Department of Computing
Imperial College
180 Queen's Gate
London SW7 2BZ

The Laplace transform of the probability distribution of the end-to-end delay in tandem networks is obtained where the first and/or second queue are G-queues, i.e, have negative arrivals. For the most general case the method is based on the solution of a boundary value problem on a closed contour in the complex plane, which itself reduces to the solution of a Fredholm integral equation of the second kind. We also consider the dependence or independence of the sojourn times at each queue in the two special cases where only one of the queues is a G-queue, the other having no negative arrivals.

1 INTRODUCTION

Negative customers were introduced into queueing systems by Gelenbe, see Gelenbe [5] for example, to represent such phenomena as killing signals in speculative parallelism, inhibitor signals in neural networks and semaphores. A negative arrival has the effect of killing an ordinary – or positive – customer in the queue, if one is present, and has no effect otherwise when the negative customer disappears. Such queues have been called *G-queues* and networks thereof *G-networks.* In Markovian networks, the choice of positive customer killed makes no difference when one is interested only in the queue length random variable. This follows from the memoryless property of the negative exponential distribution. However, this is not the case when one considers the sojourn time of a positive customer in a queue – even a single server, M/M/1 queue for which the sojourn time distribution is considered in Harrison and Pitel [6]. There are no results available for the distribution of response times (end-to-end delays) in G-networks to the authors' best knowledge. Here we

---

[1] Supported by the European Grant BRA-QMIPS of CEC DG XIII.

[2] Supported by the European Commission under ESPRIT bursary n° ERBCHBICT-920179.

† Accepted for publication in the Journal of Applied Probability; Reprinted with permission of the Applied Probability Trust.

investigate the response time random variable in an open tandem pair of G-queues with first come first served (FCFS) queueing discipline and removal of the customer at the end of the queue by negative arrivals (RCE). This is the simplest G-network, but, we will see, one with a complex solution in general.

The problem with even such a simple network is that it has *overtaking* in its most general sense. A network is only overtake-free if no customer joining a path after some special, *tagged* customer – the one being tracked – can have any influence on the time that customer takes to complete his sojourn at any server. It is *not* necessary for such a customer to actually overtake the tagged customer (which clearly would constitute 'influence'). In particular, any network with queue-length dependent service rates cannot be overtake-free since later customers affect the service rate at a node and hence can 'influence' the progress of the tagged customer. For example, Coffman et al. [1] consider a tandem pair of M/M/1 queues with processor sharing (PS) discipline at both queues or the first queue only. The influence in an open tandem G-network is fairly subtle: Positive customers arriving behind the tagged customer at either node provide a degree of 'protection' for the tagged customer from being killed by possible negative customers. Thus the progress of the tagged customer is affected – the more positive arrivals after him the more likely it is that he will not be killed.

Now, the equilibrium sojourn time distribution in a single M/M/1 G-queue has been found in Harrison and Pitel [6], and this is the required result in a tandem network for the sojourn time at the first queue only. However, because of the absence of the non-overtaking property, the sojourn time at the second queue is not independent of that at the first and so we have to proceed differently, explicitly taking into account this dependence. In fact, there is an analogy with the analysis of response times in the 3-server 'triangular' network with overtaking considered by Fayolle et al. [3]. In this network, customers taking the path of servers numbered 1-2-3 can be overtaken by customers taking the shorter path 1-2. The solution obtained for this problem considered the state of the whole network at the departure instant of the tagged customer from server 1 and its evolution in an infinitesimal interval after that. It relied on the property that the departure process from queue 1 is Poisson and the consequent Random Observer property.

We could proceed similarly as we know the departure process from the first queue is also Poisson (see Pitel [8]) but find it more convenient to model the dynamic behaviour in the first queue explicitly, conditioned only on the state seen on arrival at the whole network (at server 1). Finally, observe that there are a number of special cases of our tandem network produced by a G-queue with *no negative arrivals* in either position. In particular, if the second were such a queue, there could be no killing there and so no benefit offered by 'protection' from departures from the first queue (or externally). In other words, we can expect the sojourn time in the second queue to be independent of that in the first and we do indeed verify this. Conversely, if there were no negative arrivals at the first queue, we could not expect independence. This is in contrast with the well known observation in tandem networks of FCFS queues that the *last* queue in the series can be replaced with a more general one – e.g. M/G/1. In our case we allow the *first* to be non-standard.

To summarise, in general, negative customers destroy the independence of the sojourn times in each queue in the tandem pair that exists when there are no negative customers. Quantitatively, this complicates the analysis of the end-to-end response time which is the sum of the constituent sojourn times. When these sojourn times are independent, the distribution of their sum is just the convolution of their distributions which are known, see Reich [9]. The Laplace-Stieltjes transform is simply the product of the transforms corresponding to the individual sojourn times. However, when the sojourn times are not independent and one seeks the distribution of their sum, anything can happen – and it does! We obtain a Fredholm integral equation of the second kind – which at least has a tractable numerical solution. Hence the introduction of mathematically innocent-looking negative customers has a dramatic effect on the calculation of response time distributions, in contrast to queue length distributions in the steady state.

The remainder of the paper is organised as follows. After a brief introduction of the general model – called the G-G model – and its variants used in the sequel, we explain how we derive the distribution of the end-to-end delay in the G-G model in section 3. This leads to the boundary value problem detailed in section 3.3, whose solution yields a Fredholm integral equation of the second kind. In section 4 we consider special cases to illustrate once more the independence for a tandem pair of two normal M/M/1 queues (section 4.1) as well as that of the G-M model (section 4.2) in contrast to the conjectured dependence of the M-Gr model (section 4.3). The paper concludes in section 5.

2 DESCRIPTION OF THE SYSTEMS

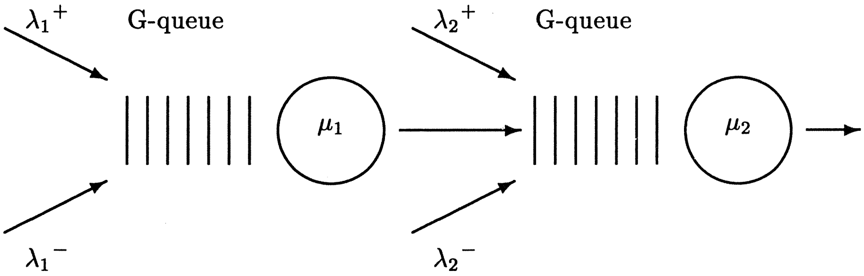
All networks considered are variants of a tandem network of two M/M/1 normal or G-queues.

2.1 The general G-G model

G-G stands for two M/M/1 G-queues in tandem as defined below. Each queue $Q_i, i \in \{1,2\}$ has an external Poisson arrival process with rate λ_i^+ representing a stream of ordinary customers, an external Poisson arrival process with rate λ_i^- representing a stream of negative customers and an exponential service time distribution with rate μ_i. The queueing discipline is FCFS for the positive customers and the killing strategy is RCE for the negative customers. There is also an internal arrival process at the second queue Q_2 representing the stream of positive customers that have completed service at the first service facility.

For a single G-queue, we use the following compact notation in the sequel: a FCFS-RCE M/M/1 G-queue with parameters (λ^+, λ^-, μ) indicates that the queue is Markovian and has FCFS ordinary arrivals with rate λ^+, negative arrivals killing the end of the queue with rate λ^- and service time distribution with rate μ.

The G-G model introduces two places in the network where a form of overtaking can occur. Indeed in each queue a customer who enters the system at time t_0, say, cannot be sure to leave it – after receiving service at the second node – before a customer arriving at time $t > t_0$ as he can be killed while waiting at one of the two queues. This can be summarized by the following figure with the abbreviation G-queue standing for M/M/1 G-queue henceforth:



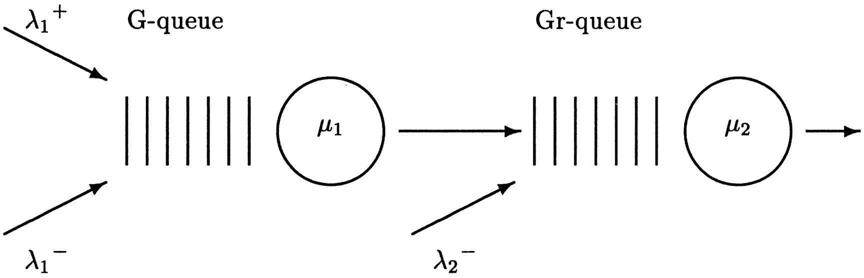
The steady state joint queue lengths probability distribution of this general tandem network Gelenbe [5] is given by

$$\begin{aligned} P[N_1 = n_1, N_2 = n_2] &= (1-\rho_1)\,\rho_1^{n_1}\,(1-\rho_2)\,\rho_2^{n_2} \qquad (1)\\ \rho_1 &= \frac{{\lambda_1}^+}{{\lambda_1}^- + \mu_1}, \quad \rho_1 < 1\\ \rho_2 &= \frac{\mu_1\,\rho_1 + {\lambda_2}^+}{{\lambda_2}^- + \mu_2}, \quad \rho_2 < 1 \end{aligned}$$

Note that $\rho_2 = \frac{P[\textit{not killed at } Q_1]\,{\lambda_1}^+ + {\lambda_2}^+}{{\lambda_2}^- + \mu_2}$.

2.2 The restricted G-Gr model

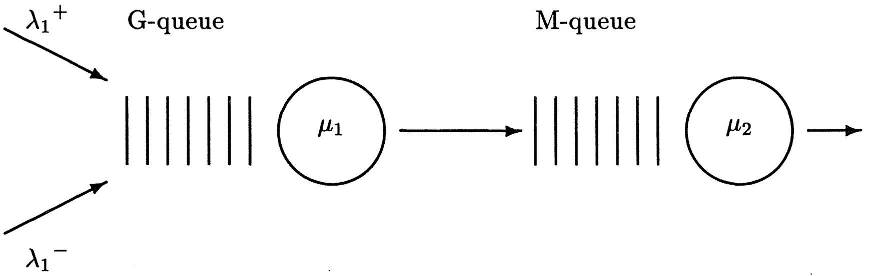
The G-Gr model also has two overtaking places but there are no positive external arrivals at the second (Gr) queue, i.e. ${\lambda_2}^+ = 0$. This is closer to the normal concept of a simple tandem network. Because of this restriction, we call the second M/M/1 G-queue a Gr-queue.



In the two following dual models, we have only one overtaking place through one M/M/1 G-queue in tandem with one normal M/M/1 queue, henceforth abbreviated to M-queue. In the first case the G-queue precedes the M-queue whereas a restricted G-queue succeeds a normal one in the second case.

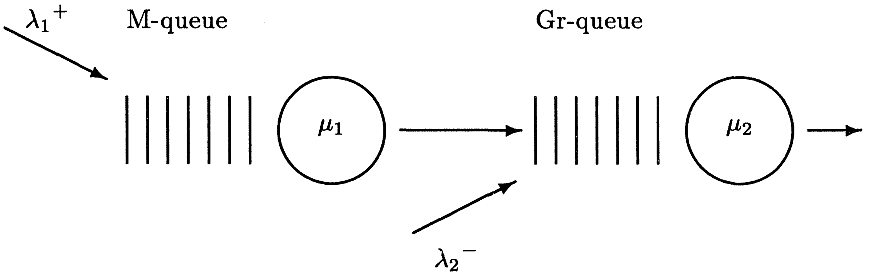
2.3 The G-M model

This corresponds to $\lambda_2{}^+ = \lambda_2{}^- = \lambda_2 = 0$.



2.4 The M-Gr model

This corresponds to $\lambda_2{}^+ = \lambda_1{}^- = 0$.



2.5 Method of solution

We call the special, monitored customer, whose sojourn time density (jointly with the probability he is not killed in the system) we seek, **the tagged customer** , which is special in the following sense. Like all customers, he obeys the specified queueing discipline *after his arrival in the facing queue*, but he may arrive at an *arbitrary position in the queue.* Of course, the tagged customer in which we are interested joins the queue according to the rules obeyed by all other customers, i.e. at the end, and we call him the **normal tagged customer**.

We will consider the normal tagged customer at arrival instants in Q_1 first and then at the time he completes service at Q_1. Because of the arrival process at Q_1 being Poisson, we use the Random Observer Property to state that initially the normal tagged customer sees the queue sizes distributed as in the steady state. Therefore we will derive two sets of dependent equations involving remaining sojourn times for the tagged customer observed (a) in Q_1 at arrival instants and (b) in Q_2 at departure instants from Q_1.

3 RESPONSE TIME DISTRIBUTION - THE G-G CASE

3.1 General formalism

Theorem 1 (G-G)

In a two server tandem network where the two queues are M/M/1 G-queues, the Laplace transform of the total sojourn time density, jointly with the probability of not being killed, is

$$W^*(s) = (1-\rho_1)\,(1-\rho_2)\,\frac{\mu_1}{{\lambda_1}^+}\,y_1(s)\,G(\rho_2,0,y_1(s),s) \tag{2}$$

where G satisfies the functional equation $(\forall x, y, z, s.\ [|x| < 1,\ |y| < 1,\ |z| < 1,\ \Re(s) > 0])$

$$R\,G = ({\lambda_1}^- + \mu_1 - \frac{{\lambda_1}^+}{z})\,G(x,y,0,s) - \frac{\mu_1 z + {\lambda_2}^+}{y}\,G(x,0,z,s) + \frac{\mu_2}{(1-y)(1-z)} \tag{3}$$

with $R = s + \lambda_1 + \lambda_2 + \mu_1 + \mu_2(1-x) - \dfrac{{\lambda_1}^+}{z} - {\lambda_1}^- z - \dfrac{{\lambda_2}^+}{y} - {\lambda_2}^- y - \mu_1 \dfrac{z}{y}$

and $y_1(s)$ the smaller root of the quadratic ${\lambda_1}^- y^2 - (s + \lambda_1 + \mu_1(1-\rho_1))y + {\lambda_1}^+ = 0$.

Proof

Let S_i be the sojourn time at Q_i, $i \in \{1,2\}$,

A_i the no. of customers *ahead of* the tagged customer in Q_i,
B_i the no. of customers *behind* the tagged customer in Q_i,
L_i the no. of cust. in Q_i seen by the tagged customer being in the other queue.

These variables are implicitly defined at arrival instants or departure instants (or any other time), which follows from the Random Observer Property of the Poisson processes involved in the analysis. We define K_{pqn} (resp. F_{pqn}) as the conditional sojourn time probability distribution in the system (resp. at Q_2), given the state of the queues described by A_1, B_1 and L_2 (resp. A_2, B_2 and L_1), and H (resp. G) as its corresponding generating function:

$$\begin{aligned}
K_{pqn}(t) &\triangleq P[S_1 + S_2 \le t | A_1 = p, B_1 = q, L_2 = n] \\
F_{pqn}(t) &\triangleq P[S_2 \le t | A_2 = p, B_2 = q, L_1 = n] \\
H(x,y,z,s) &\triangleq \sum_{p,q,n \ge 0} K^*_{pqn}(s)\; x^p\; y^q\; z^n \\
G(x,y,z,s) &\triangleq \sum_{p,q,n \ge 0} F^*_{pqn}(s)\; x^p\; y^q\; z^n
\end{aligned}$$

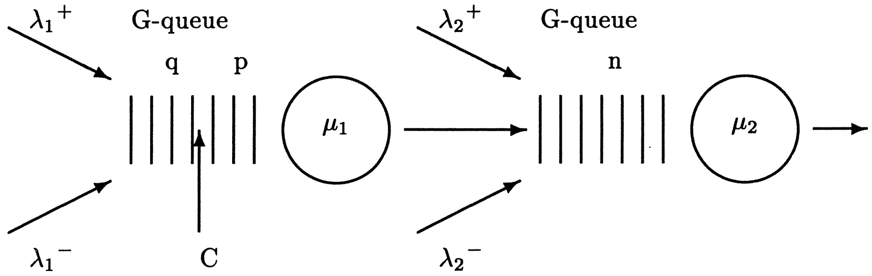


Figure 1: G-G tandem network with C in Q_1

The first step of the analysis is to pick up the tagged (ordinary) customer C in Q_1 at a random time (cf. Figure 1). In equilibrium, C faces the queue Q_1 with length distributed as A_1 on arrival (by the Random Observer Property), has no customer behind him almost surely and sees the queue Q_2 with length distributed as L_2 (in equilibrium). Hence, the Laplace transform of the response time ($S = S_1 + S_2$) density is given by

$$W^*(s) = (1-\rho_1)\,(1-\rho_2)\,H(\rho_1, 0, \rho_2, s) \tag{4}$$

with $\rho_1 = \frac{\lambda_1{}^+}{\lambda_1{}^- + \mu_1}$ and $\rho_2 = \frac{\mu_1\rho_1 + \lambda_2{}^+}{\lambda_2{}^- + \mu_2}$.

Now, the possible events that can occur during the initial small interval after the arrival $[0,h]$ for infinitesimal $h > 0$ are as follows (since all the times to the occurrences are exponential random variables):

- a positive arrival at the end of Q_1, w.p. $\lambda_1{}^+ h + o(h)$,
- a negative arrival which kills the customer at the end of Q_1, w.p. $\lambda_1{}^- h + o(h)$ (always possible as C is in Q_1),
- a departure of the customer in service at Q_1, which goes to the end of Q_2, w.p. $\mu_1 h + o(h)$ (always possible as C is in Q_1),
- a positive arrival at the end of Q_2, w.p. $\lambda_2{}^+ h + o(h)$,
- a negative arrival which kills the customer (if any) at the end of Q_2, w.p. $\lambda_2{}^- h + o(h)$,
- a departure of the customer (if any) in service at Q_2, which leaves the system, w.p. $\mu_2 1_{\{n_2>0\}}\, h + o(h)$.

No events occur w.p. $1 - (\lambda_1{}^+ + \lambda_1{}^- + \lambda_2{}^+ + \lambda_2{}^- + \mu_1 + \mu_2 1_{\{n_2>0\}})\, h + o(h)$. With similar arguments to Harrison and Pitel [6], this leads, after a few steps, to the following equation where the variable t is omitted for brevity

$$\frac{\partial K_{pqn}}{\partial t} = \lambda_1{}^+ K_{p,q+1,n} + \lambda_1{}^- \{K_{p,q-1,n}\, 1_{\{q>0\}} + 0 \cdot 1_{\{q=0\}}\}$$

$$\begin{aligned}
&+ \lambda_2{}^+ K_{p,q,n+1} + \lambda_2{}^- \{K_{p,q,n-1}\, 1_{\{n>0\}} + K_{p,q,0}\, 1_{\{n=0\}}\} \\
&+ \mu_1 \{K_{p-1,q,n+1}\, 1_{\{p>0\}} \\
&\qquad + 1 \,.\, P[S_2 \le t | A_2 = n, B_2 = 0, L_1 = q]\, 1_{\{p=0\}}\} \\
&+ \mu_2 K_{p,q,n-1}\, 1_{\{n>0\}} - (\lambda_1 + \lambda_2 + \mu_1 + \mu_2 1_{\{n>0\}})\, K_{pqn}
\end{aligned}$$

Also $k_{pqn}(0) = 0$, $\forall pqn$ (k being the derivative of K).
Taking the Laplace-Stieltjes transform of the above equation and using the generating functions H and G, with the simplified notation $H \equiv H(x,y,z,s)$ (idem for G and R below) yields

$$\begin{aligned}
s\,H \;=\; & \frac{\lambda_1{}^+}{y}[H - H(x,0,z,s)] + \lambda_1{}^- y\, H \\
& + \frac{\lambda_2{}^+}{z}[H - H(x,y,0,s)] + \lambda_2{}^-[z\, H + H(x,y,0,s)] \\
& + \mu_1\, [\frac{x}{z}\, [H - H(x,y,0,s)] + G(z,0,y,s)] + \mu_2 z\, H \\
& - [(\lambda_1 + \lambda_2{}^- + \mu_1)\, H + \mu_2\, [H - H(x,y,0,s)]]
\end{aligned}$$

In other words, we have to solve the following complex equation

$$R\,H = (\lambda_2{}^- + \mu_2 - \frac{\mu_1 x + \lambda_2{}^+}{z})\; H(x,y,0,s) - \frac{\lambda_1{}^+}{y} H(x,0,z,s) + \mu_1 G(z,0,y,s) \quad (5)$$

where $R = s + \lambda_1 + \mu_1 + \lambda_2{}^+ + (\lambda_2{}^- + \mu_2)(1-z) - \dfrac{\lambda_1{}^+}{y} - \lambda_1{}^- y - \dfrac{\mu_1 x + \lambda_2{}^+}{z}$

This equation reduces, in fact, to one in two variables y and z only as x and s behave as fixed parameters. Moreover we only need H at the particular point $(\rho_1, 0, \rho_2, s)$ which we can obtain classically as equation 5 happens to simplify when evaluated at the 'point' (ρ_1, y, ρ_2, s). This leaves us to solve

$$R(\rho_1, y, \rho_2, s)\; H(\rho_1, y, \rho_2, s) = -\frac{\lambda_1{}^+}{y} H(\rho_1, 0, \rho_2, s) + \mu_1 G(\rho_2, 0, y, s)$$

with $R(\rho_1, y, \rho_2, s) = s + \lambda_1 + \mu_1(1 - \rho_1) - \frac{\lambda_1{}^+}{y} - \lambda_1{}^- y$.
The usual argument on analyticity of the functions involved gives

$$H(\rho_1, 0, \rho_2, s) = \frac{\mu_1}{\lambda_1{}^+}\, y_1(s)\; G(\rho_2, 0, y_1(s), s) \qquad (6)$$

where $y_1(s)$ is the root involved in the time delay distribution of a FCFS-RCE M/M/1 G-queue with parameters $(\lambda_1{}^+, \lambda_1{}^-, \mu_1)$, from $R(\rho_1, y, \rho_2, s) = 0$; see Harrison and Pitel [6] for more details. As such this also implies that $y_1(s) < \rho_1$ (proof in Pitel [8]). Now to complete the theorem we derive the equation satisfied by G.

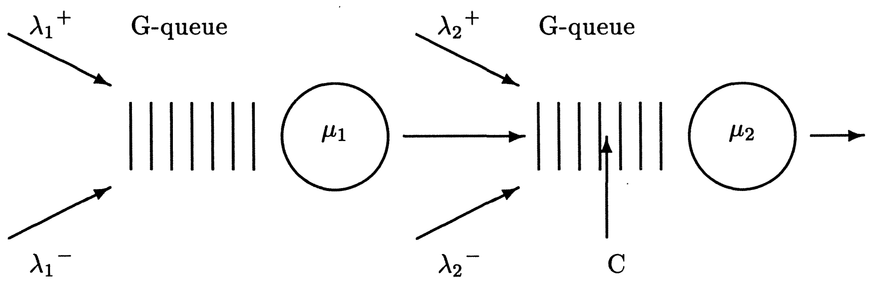


Figure 2: G-G tandem network with C in Q_2

We now consider the tagged customer at its departure instant from Q_1 and suppose (in general) he joins the second queue in any position (cf. Figure 2). Let A_2 and B_2 be the number of customers *ahead of* and *behind* the tagged customer in Q_2 respectively and L_1 the number of customers in Q_1 seen by the tagged customer at this time. The remaining sojourn time of a normal customer departing from Q_1 corresponds to the state $A_2 = N_2$, $B_2 = 0$ and $L_1 = N_1$, also interestingly and logically bound to the value of $G(x, 0, z, s)$.

Again, we consider the possible events that may occur in the small interval $[0, h]$:

- a positive arrival at the end of Q_1, w.p. $\lambda_1{}^+h + o(h)$,
- a negative arrival which kills the customer (if any) at the end of Q_1, w.p. $\lambda_1{}^-h + o(h)$
- a departure of the customer (if any) in service at Q_1, which goes to the end of Q_2, w.p. $\mu_1 \ 1_{\{n_1>0\}} \ h + o(h)$,
- a positive arrival at the end of Q_2, w.p. $\lambda_2{}^+h + o(h)$,
- a negative arrival which kills the customer at the end of Q_2, w.p. $\lambda_2{}^-h + o(h)$ (always possible as C is in Q_2),
- a departure of the customer in service at Q_2, which leaves the system, w.p. $\mu_2 h + o(h)$ (always possible as C is in Q_2).

No events occur w.p. $1 - (\lambda_1{}^+ + \lambda_1{}^- + \lambda_2{}^+ + \lambda_2{}^- + \mu_1 \ 1_{\{n_1>0\}} + \mu_2) \ h + o(h)$.

A similar derivation to the previous yields the following differential equation

$$\begin{aligned}\frac{\partial F_{pqn}}{\partial t} &= \lambda_1{}^+F_{p,q,n+1} + \lambda_1{}^-\{F_{p,q,n-1} \ 1_{\{n>0\}} + F_{p,q,0} \ 1_{\{n=0\}}\} \\ &\quad + \lambda_2{}^+F_{p,q+1,n} + \lambda_2{}^-\{F_{p,q-1,n} \ 1_{\{q>0\}} + 0 \ . \ 1_{\{q=0\}}\} \\ &\quad + \mu_1 \ F_{p,q+1,n-1} \ 1_{\{n>0\}} + \mu_2\{F_{p-1,q,n} \ 1_{\{p>0\}} + 1 \ .1_{\{p=0\}}\} \\ &\quad - (\lambda_1 + \lambda_2 + \mu_1 \ 1_{\{n>0\}} + \mu_2) \ F_{pqn}\end{aligned}$$

with $f_{0,q,n}(0) = \mu_2$ (f denotes the derivative of F).
Laplace transforming it then gives

$$\begin{aligned} sG \; - \; \frac{\mu_2}{(1-y)(1-z)} &= \\ & \frac{{\lambda_1}^+}{z}[G - G(x,y,0,s)] + {\lambda_1}^-[zG + G(x,y,0,s)] \\ & + \frac{{\lambda_2}^+}{y}[G - G(x,0,z,s)] + {\lambda_2}^- y \, G + \frac{\mu_1 z}{y}[G - G(x,0,z,s)] + \mu_2 \; x \; G \\ & -[(\lambda_1 + \lambda_2 + \mu_2) \; G + \mu_1 \; [G - G(x,y,0,s)]] \end{aligned}$$

Rearranging then produces the sought functional equation for G on the given domain of the variables x, y, z, s. □

This equation remains to be solved over the unit disks in the x, y, z planes, where G is by definition analytic. Unfortunately there is no short cut to a straightforward solution as the term ${\lambda_1}^- + \mu_1 - \frac{{\lambda_1}^+}{z}$ disappears only for $z = \rho_1$ and not for $z = y_1(s)$, $s \neq 0$. This means we have to evaluate G for all arbitrary complex values of z, just to get G at $(\rho_2, 0, y_1(s), s)$. We do this in section 3.3.

3.2 Probability of not being killed in the system

Before embarking on the main problem we can verify the consistency of our equations by computing the probability of a customer not being killed while sojourning in the system. This probability is $W^*(0)$.

Corollary 1 (G-G)

The probability of not being killed while sojourning in the system is

$$W^*(0) = \frac{\mu_1}{{\lambda_1}^- + \mu_1} \frac{\mu_2}{{\lambda_2}^- + \mu_2}$$

Proof

The following relations stand

$$\begin{aligned} W^*(0) &= (1-\rho_1)\,(1-\rho_2)\,H(\rho_1,0,\rho_2,0) \\ H(\rho_1,0,\rho_2,0) &= \frac{\mu_1}{{\lambda_1}^+} y_1(0)\; G(\rho_2,0,y_1(0),0) \\ y_1(0) &= \rho_1 \\ G(\rho_2,0,\rho_1,0) &= \frac{\mu_2}{{\lambda_2}^- + \mu_2} \frac{1}{(1-\rho_1)(1-\rho_2)} \end{aligned} \qquad (7)$$

as equation 3 simplifies at $(\rho_2, y, \rho_1, 0)$ as follows. The first term on the r.h.s. of equation 3 vanishes and leads to a well-known tractable problem, which is to solve

$$R(\rho_2,y,\rho_1,0)\; G(\rho_2,y,\rho_1,0) = -\frac{\mu_1\rho_1 + {\lambda_2}^+}{y}\; G(\rho_2,0,\rho_1,0) + \frac{\mu_2}{1-\rho_1} \frac{1}{1-y}$$

for $|y| < 1$ where $R(\rho_2, y, \rho_1, 0) = (y - \rho_2)\,(\frac{\lambda_2{}^- + \mu_2}{y} - \lambda_2{}^-)$. There is therefore only one root of the equation $R(\rho_2, y, \rho_1, 0) = 0$, viz $y = \rho_2$, in the unit disk, in which $G(\rho_2, y, \rho_1, 0)$ (a function of y) has to be analytic. This produces the above equation 7 and hence the result for $W^*(0)$. □

This probability is simply the product of the probability of not being killed in queue 1 and that in queue 2 – these are therefore independent.

3.3 A Riemann-Hilbert-Carleman boundary value problem

The general solution for the G-G case is given by the generating function G which is itself the solution of a Riemann-Hilbert-Carleman (RHC) boundary value problem. This is given by the following.

Proposition 1

The functional equation 3, i.e.

$$RG = (\lambda_1{}^- + \mu_1 - \frac{\lambda_1{}^+}{z})G(x, y, 0, s) - \frac{\mu_1 z + \lambda_2{}^+}{y}G(x, 0, z, s) + \mu_2 \frac{1}{1-y}\frac{1}{1-z}$$

where $R = s + \lambda_1 + \lambda_2 + \mu_1 + \mu_2(1-x) - \frac{\lambda_1{}^+}{z} - \lambda_1{}^- z - \frac{\lambda_2{}^+}{y} - \lambda_2{}^- y - \mu_1 \frac{z}{y}$

is equivalent to a boundary problem of type Riemann-Hilbert-Carleman, namely to find a function H satisfying

$$H(z) - H(\overline{z}) = h(z) \qquad z \in L$$

where h is given and L is a closed contour inside the unit disk. In terms of G, H is defined by equation 11. The solution for H reduces to a Fredholm integral equation of the second kind, which has to be solved numerically in general.

The proof of the proposition is in three parts which are given in the following subsections that identify the functions H and h as well as the contour L and describe the solution technique in the Fredholm integral.

3.3.1 Analyticity for a general solution for G

Firstly, we recall that x and s can be treated as fixed parameters in equation 3 and therefore G as a function of only two variables y and z. We rewrite it as $\tilde{G}(y, z) \equiv G(x, y, z, s)$. We note that at most we need $\tilde{G}(0, z)$, which of course does still depend on x and s.

We shall use the analyticity of G, which by definition holds inside the unit disks in the x, y, z planes, with $\Re(s) \geq 0$ – we call $\Re_C$ this region of convergence. When $R = 0$ in $\Re_C$ and $\tilde{G}(y, z)$ is finite, the r.h.s. of equation 3 must also vanish. R is quadratic w.r.t. both y and z. Let $z = z(y)$ and $z^\sigma = z^\sigma(y)$ be the two roots of $R = 0$ w.r.t. z, i.e. the roots of

$$(\lambda_1{}^- y + \mu_1)z^2 - [(\alpha - \lambda_2{}^- y)y - \lambda_2{}^+]\, z + \lambda_1{}^+ y = 0 \qquad (8)$$

with $\quad \alpha = s + \lambda_1 + \lambda_2 + \mu_1 + \mu_2(1-x)$

The presence of λ_2^- increases the order of the two-variable y, z polynomial (8) in y and should complicate things. The superscript σ can be regarded as mapping from one solution to the other. Then the right-hand side of equation 3 also equals 0 at points $(y, z(y))$ and $(y, z^\sigma(y))$ when those points are within the region of convergence $\Re_C$ of $\tilde{G}(y,z)$. This yields

$$(\lambda_1^- + \mu_1 - \frac{\lambda_1^+}{z})\ \tilde{G}(y,0) - \frac{\mu_1 z + \lambda_2^+}{y}\ \tilde{G}(0,z) + \frac{\mu_2}{1-y}\ \frac{1}{1-z} = 0 \qquad (9)$$

$$(\lambda_1^- + \mu_1 - \frac{\lambda_1^+}{z^\sigma})\ \tilde{G}(y,0) - \frac{\mu_1 z^\sigma + \lambda_2^+}{y}\ \tilde{G}(0,z^\sigma) + \frac{\mu_2}{1-y}\ \frac{1}{1-z^\sigma} = 0$$

In the situation when these two equations hold simultaneously, we can eliminate $\tilde{G}(y,0)$ between them. This gives

$$(z^\sigma - \rho_1)\ (\mu_1 z + \lambda_2^+)\ z\ \tilde{G}(0,z) - (z - \rho_1)\ (\mu_1 z^\sigma + \lambda_2^+)\ z^\sigma\ \tilde{G}(0,z^\sigma) = h(z)(10)$$

where $h(z) = \mu_2 \dfrac{y(z)}{1-y(z)}\ [\dfrac{z}{1-z}\ (z^\sigma - \rho_1) - \dfrac{z^\sigma}{1-z^\sigma}\ (z-\rho_1)]\ z, z^\sigma \in \Re_C$

Bearing in mind that

$$\begin{aligned} zz^\sigma &= \frac{\lambda_1^+ y}{\lambda_1^- y + \mu_1} \\ z + z^\sigma &= \frac{(\alpha - \lambda_2^- y)y - \lambda_2^+}{\lambda_1^- y + \mu_1} \end{aligned}$$

$$y(z) = \frac{\alpha z - \lambda_1^+ - \lambda_1^- z^2 \pm \sqrt{(\alpha z - \lambda_1^+ - \lambda_1^- z^2)^2 - 4\lambda_2^- z(\lambda_2^+ + \mu_1 z)}}{2\lambda_2^- z}$$

where the sign of the square root is chosen so that $|y(z)| \leq |y^\sigma(z)|$ noticing that $y(0) = 0$ (by solving equation 8 as a quadratic in y), we introduce

$$H(z) = (\frac{\lambda_1^+ y(z)}{\lambda_1^- y(z) + \mu_1} - \rho_1 z)\ (\mu_1 z + \lambda_2^+)\ \tilde{G}(0,z) \qquad z \in \Re_C \qquad (11)$$

where λ_2^- is implicitly present through $y(z)$. This finally reduces equation 10 to

$$H(z) - H(z^\sigma) = h(z) \qquad z, z^\sigma \in \Re_C \qquad (12)$$

where we recall that z and z^σ are the roots of $R = 0$ regarded as a quadratic in z.

3.3.2 Branch points and introduction of contours L and L_e

The solutions of the quadratic equation 8 in z introduce branch cuts in the y-plane which map to contours in the z-plane defining L of proposition 1 and L_e. The properties of the branch cuts are given in the following lemma.

Lemma 1

1. $z(y)$ and $z^\sigma(y)$ have four real branch points y_i, $i \in \{1,2,3,4\}$ satisfying

$$0 < y_1 < a_1 < y_2 < 1 < y_3 < a_2 < y_4$$

where a_1, a_2 are the roots of the equation $b(y) = 0$

$$\text{where} \qquad b(y) = \alpha y - {\lambda_2}^- y^2 - {\lambda_2}^+$$

and the y_i $(i \in \{1,2,3,4\})$ are the roots of the equation $\delta(y) = 0$

$$\text{where} \qquad \delta(y) = b(y)^2 - 4{\lambda_1}^+ y\,({\lambda_1}^- y + \mu_1)$$

2. $z^\sigma(y) = \overline{z}(y)$ for $y \in [y_1, y_2] \cup [y_3, y_4]$
3. $\forall y, |z(y)| \leq |z^\sigma(y)|$,
   i.e. $z(y)$ always lies inside L_e whereas $z^\sigma(y)$ always lies outside L.
4. There is one and only one root $z(y)$ of $R = 0$ with $|z(y)| \leq 1$ when $|y| = 1$.
5. The corresponding properties hold for $y(z)$.

Proof

1. The discriminant of the quadratic equation 8 for $z(y)$ is $\delta(y)$.

   Henceforth we will assume that x $(= \rho_2)$ and s are real, which is the domain of interest for our problem. Of course, we still demand that our parameters lie inside $\Re_C$, i.e. $x \in [-1,1]$ and $s \geq 0$. This restriction implies that we are now looking for the roots of a complex polynomial of degree 4 with real coefficients. Then, if a complex root is found, its conjugate will also be a solution. As such, this polynomial has either zero, two or four real roots. It is easy to verify that $b(0) = -{\lambda_2}^+ < 0$, $b(1) = \alpha - \lambda_2 > 0$ and $b(+\infty) = -\infty < 0$ so that there exist two reals $a_1 \in]0,1[$ and $a_2 \in]1,+\infty[$ s.t. $b(a_1) = b(a_2) = 0$. Then for δ itself we have $\delta(0) = {\lambda_2}^{+2} > 0$, $\delta(a_1) < 0$, $\delta(1) = (s+\mu_2(1-x))^2+(\mu_1+{\lambda_1}^- - {\lambda_1}^+)^2 > 0$, $\delta(a_2) < 0$ and $\delta(+\infty) = +\infty > 0$. This completes the first part of the lemma. Note that if ${\lambda_2}^+ = 0$ then $y_1 = a_1 = 0$ as $\frac{\delta(y)}{y} < 0$ at $y = 0$.

2. Obviously $\delta(y) \leq 0$ when y falls into one of the two real intervals $[y_1, y_2]$ and $[y_3, y_4]$. Then the two solutions of the quadratic equation 8 are complex conjugate: $z^\sigma(y) = \overline{z}(y)$. As the denominator ${\lambda_1}^- y + \mu_1$ of $z(y)$ cannot vanish for y in either interval, $z(y)$ is a continuous function of y and

maps a curve in the right upper quadrant plane ($\Re(z) \geq 0, \Im(z) \geq 0$). By symmetry w.r.t. the real axis, $z^\sigma(y)$ maps a symmetric curve in the right lower quadrant ($\Re(z) \geq 0, \Im(z) \leq 0$). In other words, when $y \in [y_1, y_2]$ ($[y_3, y_4]$ respectively), the quadratic equation maps a closed contour L (L_e respectively), symmetric w.r.t. the real axis. Moreover, we note that

$$|z(y)|^2 = z(y)\overline{z}(y) = \frac{\lambda_1{}^+ y}{\lambda_1{}^- y + \mu_1}$$

which is an increasing function of y. Let y' be in $[y_1, y_2]$, y'' in $[y_3, y_4]$ so that we have $y' < 1 < y''$ and so $\frac{\lambda_1{}^+ y'}{\lambda_1{}^- y' + \mu_1} \leq \frac{\lambda_1{}^+}{\lambda_1{}^- + \mu_1} \leq \frac{\lambda_1{}^+ y''}{\lambda_1{}^- y'' + \mu_1}$ or $|z(y')| \leq |z(y'')|$ which means that $L \subseteq L_e$ and $|z(y')| \leq 1$ from the stability condition $\rho_1 < 1$. Therefore L lies inside the unit disk and inside L_e (which is not necessarily in the unit disk). We shall call $\Re_L$ ($\Re_{L_e}$ resp.) the region inside L (L_e resp.) and $\Re_{L_e \backslash L}$ the ring-shaped area between L and L_e. Of course, $\Re_L \subset \Re_C$. Note that this is important for the issue of our problem, but that for the complex analysis itself what matters is only that $L \subset L_e$.

3. Let us consider the function $r(y) = \frac{z(y)}{z^\sigma(y)}$. It is analytic everywhere in the complex plane cut along $[y_1, y_2]$ and $[y_3, y_4]$. For values of y other than those on the unit circle, $z(y)$ and $z^\sigma(y)$ are determined by analytic continuation, wherever possible (see Pitel [8] for more details). Now, on boundaries we have:
   - $\forall y \in [y_1, y_2] \cup [y_3, y_4]$, $z^\sigma(y) = \overline{z(y)}$, therefore $|r(y)| = 1$,
   - at infinity, $z(y) \sim \frac{1}{y}$ and $z^\sigma(y) \sim \frac{\lambda_1{}^+ y}{\lambda_1{}^-}$, i.e. $\lim_{y \to \infty} r(y) = 0$.

   Using the maximum modulus theorem, the maxima of the function $|r(y)|$ are reached on the boundaries and so we have $\forall y, |r(y)| \leq 1$.
   Similarly the maximum of $|z(y)|$ is reached on L_e and the minimum of $|z^\sigma(y)|$ on L. So $z(y)$ always lies in $\Re_{L_e}$ and $z^\sigma(y)$ outside $\Re_L$. Hence $|z(y)| \leq \sqrt{\frac{\lambda_1^+ y_4}{\lambda_1^- y_4 + \mu_1}}$ whereas $|z^\sigma(y)| \geq \sqrt{\frac{\lambda_1^+ y_1}{\lambda_1^- y_1 + \mu_1}}$.

4. Let us take $f(z) = yz[s + \lambda_1 + \mu_1 + \mu_2(1-x) + \lambda_2{}^-(1-y) + \lambda_2{}^+(1-\frac{1}{y})]$ and $g(z) = -(\lambda_1{}^+ + \lambda_1{}^- z^2)y - \mu_1 z^2$. It is easy to see that for $|y| = |z| = 1$, $|f(z)| > \lambda_1 + \mu_1$ and $|g(z)| \leq \lambda_1 + \mu_1$. Therefore $|f(z)| > |g(z)|$ on $|y| = |z| = 1$. Using Rouche's theorem, $f(z)$ and $f(z) + g(z)$ have the same numbers of zeros inside this domain. As the only zero for $f(z)$ is 0, $f(z) + g(z)$ has only one and hence so has $R(x, y, z, s)$. Therefore there is only one root $z(y)$ of R s.t. $|z(y)| \leq 1$ when $|y| = 1$.

5. An analogous proof could be made for $y(z)$ with 4 branch points s.t. $0 < z_1 < z_2 < 1 < z_3 < z_4$. Also $[z_1, z_2]$ (resp. $[z_3, z_4]$) is mapped by $y(z)$ and $y^\sigma(z)$ onto a closed curve M (resp. M_e) in the y-plane and $M \subset M_e$. An important property could be used in the sequel: the domain inside

M cut along $[y_1, y_2]$ is conformally mapped onto the domain inside L cut along $[z_1, z_2]$ by the transformation $y \to z(y)$.

□

Note that when s approaches infinity, so do y_3, y_4, z_3 and z_4, whereas y_1, y_2, z_1 and z_2 approach 0. Then $L \to 0$, $L_e \to D(0, \sqrt{\frac{\lambda_1^+}{\lambda_1^-}})$, $M \to D(0, \sqrt{\frac{\lambda_2^+}{\lambda_2^-}})$ and $M_e \to 0$.

3.3.3 Solution to the boundary value problem

From lemma 1, the function G is given by the function H which is a solution of the following RHC boundary value problem: we seek a function $H(z)$, analytic inside the closed region $\Re_L$, given that on the boundary L of that region it satisfies

$$H(z) - H(\overline{z}) = h(z) \qquad z \in L$$

It is known that the solution is given by

$$H(z) = \frac{1}{2\pi i} \int_L \frac{\gamma(\overline{u})}{u - z}\, du + C \qquad z \in \Re_L \tag{13}$$

where C is a constant, $\gamma(u)$ is defined on L and satisfies

$$1.\ \gamma(u) + \gamma(\overline{u}) = 0 \tag{13.1}$$

$$2.\ \gamma(u) + \tfrac{1}{2\pi i} \int_L \left(\tfrac{dv}{v-u} - \tfrac{d\overline{v}}{\overline{v}-\overline{u}} \right) \gamma(v) = h(u) \tag{13.2}$$

H is thereby expressed as a Cauchy-type integral, where the density of the integral is obtained as the solution of a Fredholm integral equation in equation 13.2 (see Gakhov [4]), which is unique to within an additive constant. We first solve this integral equation involving $\gamma(u)$ to obtain $H(z)$. By a change of variables $v = z(t)$, $u = z(y)$, we write $\beta(y) = \gamma(z(y))$, $g(y) = h(z(y))$ and go round L clockwise from $z(y_1)$ to $z(y_2)$ with $z(t), \beta(t)$ and then from $z(y_2)$ to $z(y_1)$ with $\overline{z}(t), -\beta(t)$ according to 13.1. Then 13.2 becomes

$$\beta(y) + \frac{1}{2\pi i} \int_{y_1}^{y_2} \frac{\partial}{\partial t} \ln \frac{[z(t) - z(y)]\,[\overline{z}(t) - z(y)]}{[z(t) - \overline{z}(y)]\,[\overline{z}(t) - \overline{z}(y)]}\, \beta(t)\, dt = g(y) \tag{14}$$

with β defined for $y \in [y_1, y_2]$ as $\gamma(u)$ defined on L implies $z(y) \in L$.

After some manipulations, letting

$$f_{z(y)}(t) = ({\lambda_1}^- t + \mu_1)\,({\lambda_1}^- y + \mu_1)\,[z(t) - z(y)]\,[\overline{z}(t) - z(y)]$$

and bearing in mind that $z(t)$ and $z(y)$ satisfy equation 8, we have

$$\begin{aligned} f_{z(y)}(t) \;=\; & (t - y)\,[{\lambda_1}^+ \mu_1 + z(y)\,[{\lambda_2}^- t({\lambda_1}^- y + \mu_1) \\ & + \mu_1({\lambda_2}^- y - \alpha) - {\lambda_1}^- {\lambda_2}^+]] \end{aligned}$$

$$\begin{aligned}
f'_{z(y)}(t) &= \lambda_1{}^+\mu_1 + z(y)\,[2\lambda_2{}^-t(\lambda_1{}^-y+\mu_1) \\
&\qquad\qquad - \mu_1\alpha - \lambda_1{}^-\lambda_2{}^-y^2 - \lambda_1{}^-\lambda_2{}^+] \\
f'_{z(y)}f_{\overline{z}(y)} &- f'_{\overline{z}(y)}f_{z(y)} = \lambda_1{}^+\mu_1\lambda_2{}^-(\lambda_1{}^-y+\mu_1)[z(y)-\overline{z}(y)](t-y)^2 \\
f_{z(y)}f_{\overline{z}(y)} &= \lambda_1{}^+(t-y)^2[\lambda_1{}^+\mu_1{}^2+(\lambda_2{}^-yt-\lambda_2{}^+) \\
&\qquad [\lambda_2{}^-t(\lambda_1{}^-y+\mu_1)+\mu_1(\lambda_2{}^-y-\alpha)-\lambda_1{}^-\lambda_2{}^+]]
\end{aligned}$$

This simplifies equation 14 ($y \in [y_1, y_2]$) to

$$\beta(y) + \frac{1}{2\pi i}\,\mu_1\lambda_2{}^-(\lambda_1{}^-y+\mu_1)[z(y)-\overline{z}(y)]\int_{y_1}^{y_2} K(y,t)\,\beta(t)\,dt = g(y) \qquad (15)$$

where

$$\begin{aligned}
K(y,t) &= 1/[\lambda_1{}^+\mu_1{}^2 + (\lambda_2{}^-yt - \lambda_2{}^+)\,[\lambda_2{}^-t(\lambda_1{}^-y+\mu_1) \\
&\qquad + \mu_1(\lambda_2{}^-y-\alpha) - \lambda_1{}^-\lambda_2{}^+]] \\
z(y) &= \frac{\alpha y - \lambda_2{}^-y^2 - \lambda_2{}^+ \pm i\sqrt{4\lambda_1{}^+(\lambda_1{}^-y+\mu_1) - (\alpha y - \lambda_2{}^-y^2 - \lambda_2{}^+)^2}}{2(\lambda_1{}^-y+\mu_1)} \\
g(y) &= \mu_2\,\frac{y}{1-y}\,[\frac{z(y)}{1-z(y)}\,(z^\sigma(y)-\rho_1) - \frac{z^\sigma(y)}{1-z^\sigma(y)}\,(z(y)-\rho_1)]
\end{aligned}$$

and hence to

$$\begin{aligned}
\beta(y) &- \frac{\mu_1\lambda_2{}^-}{\pi}\sqrt{4\lambda_1{}^+(\lambda_1{}^-y+\mu_1) - (\alpha y - \lambda_2{}^-y^2 - \lambda_2{}^+)^2}\int_{y_1}^{y_2} K(y,t)\,\beta(t)\,dt \\
&= g(y)
\end{aligned}$$

where $y \in [y_1, y_2]$.

Having solved this Fredholm integral equation of the second kind numerically, one can substitute the resulting function $\beta(y)$ into equation 13, which can be rewritten as

$$H(z) = -\frac{1}{2\pi i}\int_{y_1}^{y_2}\left(\frac{z'(y)}{z(y)--z} + \frac{\overline{z}'(y)}{\overline{z}(y)-z}\right)\beta(y)\,dy + C \qquad z \in \Re_L \quad (16)$$

where $z(y)$ is defined in equation 15.

It remains to find the constant $C = C(\rho_2, s)$ and do the integration, again numerically since β is only determinable numerically (to our best knowledge).

Let y_0 be s.t. $R(\rho_2, y_0, \rho_1, s) = 0$ in $\Re_C$. It is easy to prove that y_0 is the smaller root involved in the time delay distribution of a FCFS-RCE M/M/1 G-queue with parameters ($\mu_1\rho_1 + \lambda_2{}^+$, $\lambda_2{}^-$, μ_2). Therefore

$$\begin{aligned}
y_0 &= [s + \rho_1\mu_1 + \lambda_2 + \mu_2(1-\rho_2) - \\
&\qquad \sqrt{(s + \rho_1\mu_1 + \lambda_2 + \mu_2(1-\rho_2))^2 - 4\lambda_2{}^-(\mu_1\rho_1 + \lambda_2{}^+)}]/(2\lambda_2{}^-)
\end{aligned}$$

which also satisfies $y_0 < \rho_2$. Note that $y_0 \notin (y_1, y_2)$ as $z_0 \overset{\mathrm{def}}{=} \rho_1$ is real. Now, if $\frac{\lambda_1^+ y_0}{\lambda_1^- y_0 + \mu_1}\,\frac{1}{\rho_1} > \rho_1$, i.e. ρ_1 is indeed the smaller root, we have $z(y_0) = \rho_1 = z_0$, otherwise $z^\sigma(y_0) = \rho_1 = z_0$.

Also $\rho_1 \in \Re_{L_e} \cap \Re_C$. Indeed for $y \in [y_3, y_4] \subset [1, y_4]$, $\rho_1^2 < \rho_1 = \frac{\lambda_1^+}{\lambda_1^- + \mu_1} \leq \frac{\lambda_1^+ y}{\lambda_1^- y + \mu_1} = |z(y)|^2$ so that $\rho_1 < |z(y)|$ and so ρ_1 is always inside L_e.

Now, whatever the value of z_0, the point (ρ_2, y_0, ρ_1, s) lies in $\Re_C$ and therefore satisfies equation 9. This yields

$$\tilde{G}(0, \rho_1) = \frac{\mu_2}{(1-\rho_1)(\mu_1 \rho_1 + \lambda_2{}^+)} \frac{y_0}{1 - y_0} = \frac{W_2^*(s)}{(1-\rho_1)(1-\rho_2)} \tag{17}$$

where $W_2^*(s)$ is the time delay distribution of a FCFS-RCE M/M/1 G-queue with parameters $(\mu_1 \rho_1 + \lambda_2{}^+,\ \lambda_2{}^-,\ \mu_2)$, according to the observation made earlier on y_0. Then by definition of $H(z)$ in $\Re_C$, equations 17 and 11 give

$$\begin{aligned} H(\rho_1) &= (\frac{\lambda_1{}^+ y_0}{\lambda_1{}^- y_0 + \mu_1} - \rho_1^2)\ (\mu_1 \rho_1 + \lambda_2{}^+)\ \tilde{G}(0, \rho_1) \\ &= (\frac{\lambda_1{}^+ y_0}{\lambda_1{}^- y_0 + \mu_1} - \rho_1^2)\ \frac{\mu_2}{1-\rho_1}\ \frac{y_0}{1-y_0} \end{aligned}$$

Moreover, $H(z)$ is also given by equation 16 in $\Re_L$. Therefore, if ρ_1 lies in $\Re_L$ (i.e. $z_0 = z(y_0)$), we can use equation 16 straightaway to determine C:

$$C = H(\rho_1) + \frac{1}{2\pi i} \int_{y_1}^{y_2} \left(\frac{z'(y)}{z(y) - \rho_1} + \frac{\overline{z}'(y)}{\overline{z}(y) - \rho_1} \right) \beta(y)\, dy \tag{18}$$

But if ρ_1 lies in $\Re_{L_e \setminus L} \cap \Re_C$ (i.e. $z_0 = z^\sigma(y_0)$), we need to proceed differently. $H(\rho_1)$ is still given by equation 11, which is valid inside $\Re_C$ as opposed to equation 16. But when $z^\sigma(y_0)$ lies in $\Re_{L_e \setminus L}$, $z(y_0)$ lies in $\Re_L$, and then $H(z)$ given by equation 16 is valid at $z(y_0)$. So we use equation 12 to link $H(z^\sigma(y_0))$ and $H(z(y_0))$:

$$C = H(\rho_1) + h(z(y_0)) + \frac{1}{2\pi i} \int_{y_1}^{y_2} \left(\frac{z'(y)}{z(y) - z(y_0)} + \frac{\overline{z}'(y)}{\overline{z}(y) - z(y_0)} \right) \beta(y)\, dy \tag{19}$$

recalling the relation $z(y_0) = \frac{\lambda_1^+ y_0}{(\lambda_1^- y_0 + \mu_1)\rho_1}$.

Note that some numerical examples with Mathematica prove that both cases do occur. For instance, with all parameters $(\lambda_1{}^+, \lambda_1{}^-, \lambda_2{}^+, \lambda_2{}^-, \mu_1, \mu_2)$ equal to 1, if $s = 0$ then ρ_1 lies in $\Re_L$ whereas if $s = 5$ ρ_1 lies in $\Re_{L_e \setminus L} \cap \Re_C$.

3.3.4 Computation of quantiles

We now have all we need to find an expression for $G(\rho_2, 0, z, s)$ from which (the Laplace transform of) the required density function follows.

$$G(\rho_2, 0, z, s) = \tilde{G}(0, z) = \frac{H(z)}{(z^\sigma - \rho_1)\ z\ (\mu_1 z + \lambda_2{}^+)} \qquad z \in \Re_C$$

from equation 11. We identify z^σ after solving $R(\rho_2, y, z, s) = 0$, where y is the smaller root and $z^\sigma = \frac{\lambda_1^+ y}{(\lambda_1^- y + \mu_1) z}$. Then, if

- $z \in \Re_L$, we use equation 16 for $H(z)$.
- $z \in \Re_{L_e \setminus L} \cap \Re_C$, we use the same trick that we used to compute the constant C. That is, we treat z as being $z^\sigma(y)$ and use the relation between z and $z^\sigma(y)$ to first get the value for $z(y)$ which lies inside $\Re_L$ and satisfies equation 16. Then we come back to $z^\sigma(y)$ through equation 12.
- $z \in \overline{\Re_{L_e}} \cap \Re_C$, we would analytically continue the function, but we do not pursue this as it is not of our domain of interest.

In fact, as we are looking for $G(\rho_2, 0, y_1(s), s)$ we just need to consider in which of these intervals $y_1(s)$ falls. Two earlier remarks – $y_1(s) < \rho_1$ and ρ_1 always inside L_e – prove that $y_1(s)$ is also always inside $\Re_{L_e}$.

The required Laplace transform follows as

$$W^*(s) = \frac{(1-\rho_1)(1-\rho_2)\mu_1 y_1(s) G(\rho_2, 0, y_1(s), s)}{\lambda_1^+}$$

and to obtain quantiles of the time delay distribution, numerical inversion is used Dubner and Abate [2].

Finally we summarize the numerical steps in our algorithm:

- Find the values of y_1 and y_2 to compute the integral required for C and H.
- Compute the constant $C(\rho_2, s)$ given by equation 18 or 19, depending on the value of s. We notice that ρ_1 rapidly falls into the region $\Re_{Le \setminus L}$ as s increases, so that $C(\rho_2, s)$ is predominantly given by equation 19.
- Find the solution γ of the Fredholm integral equation 15.
- Identify in which interval $y_1(s)$ falls – i.e. a) solve $R(\rho_2, y_s, y_1(s), s) = 0$ where y_s is the smaller root of the quadratic in y, b) if $\frac{\lambda_1^+ y_s}{(\lambda_1^- y_s + \mu_1) y_1(s)} > y_1(s)$, then $y_1(s)$ is inside L otherwise is in $\Re_{Le \setminus L}$ – to evaluate $\tilde{G}(0, y_1(s))$. Again recall that we noticed that $y_1(s)$ quickly falls inside L as s increases. So $H(y_1(s))$ is generally given by equation 16.
- Invert the Laplace transform.

$\square$

In the appendix we give some numerical examples of L, L_e and the positions of ρ_1 and $y_1(s)$.

3.4 An alternative method

An alternative approach would be to use a similar method to that of Fayolle et al. [3] which considered response times in a 3-node network with overtaking. Only one complex relation need be derived, involving the remaining sojourn time for the tagged customer in Q_2 at departure instants from Q_1. The first part of the method is based on the fact that the departure process from Q_1

is Poisson (cf. Pitel [8]). We can then use the Random Observer Property to state that the queue size faced by the normal tagged customer at the second queue (N_2) is independent of the number of customers left behind at the first queue (N_1). This leads to

Theorem 2

In a two server tandem network of M/M/1 G-queues with parameters (${\lambda_1}^+$, ${\lambda_1}^-$, μ_1) and (${\lambda_2}^+, {\lambda_2}^-, \mu_2$) respectively, the Laplace transform of the sojourn time density, jointly with the probability of not being killed, is

$$W^*(s) = \sum_{n_1 \geq 0} \phi(n_1, s) \int_0^{\infty} e^{-st} P_{n_1}(t) f_{S_1}(t)\, dt$$

where $P_{n_1}(t)$ is given by the transient solution of an M/M/1 queue (${\lambda_1}^+$, ${\lambda_1}^-$), $f_{S_1}(t)$ is the sojourn time at a FCFS-RCE M/M/1 G-queue (${\lambda_1}^+$, ${\lambda_1}^-$, μ_1) and $\phi(n_1, s)$ is given by the series $(1-\rho_2)G(\rho_2, 0, z, s) = \sum_{n_1} z^{n_1}\phi(n_1, s)$ with G satisfying the relation 3.

The proof may be found in Pitel [8].

4 SPECIAL CASES

We now consider various special cases of our tandem network, and consider independence properties of the two queues in particular.

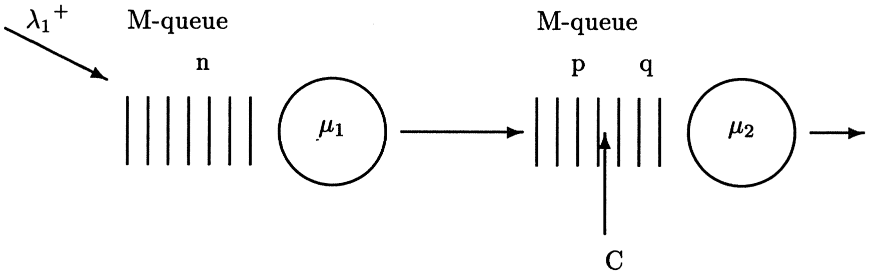
4.1 The classical tandem network

Corollary 2 (M-M)

In a classical tandem network where both queues are M/M/1 queues, the sojourn times in each queue are independent and distributed as for corresponding queues in isolation.

Proof

This network corresponds to the following situation:



${\lambda_2}^+ = {\lambda_2}^- = \lambda_2 = {\lambda_1}^- = 0$

$$\lambda_1 = \lambda_1{}^+,\ \rho_1 = \frac{\lambda_1}{\mu_1},\ \rho_2 = \frac{\lambda_1}{\mu_2},\ \lambda_1 < min(\mu_1, \mu_2)$$

Theorem G-G reduces to the problem of solving the equation satisfied by G. Knowing the independence property of this network, we guess the solution for G,

$$G(x, y, z, s) = \frac{1}{1-y}\,\frac{1}{1-z}\,\frac{\mu_2}{\mu_2(1-x)+s}$$

and just check that it verifies equation 3. It clearly expresses the non-influence of the first queue (represented through the variable z) and of the number of customers queueing in the second one behind C (represented through y) on the remaining sojourn time at the second queue (represented through x).

Next we compute $y_1(s)$ from its quadratic equation which degenerates to a simple linear equation after setting $\lambda_1{}^-$ to 0, giving

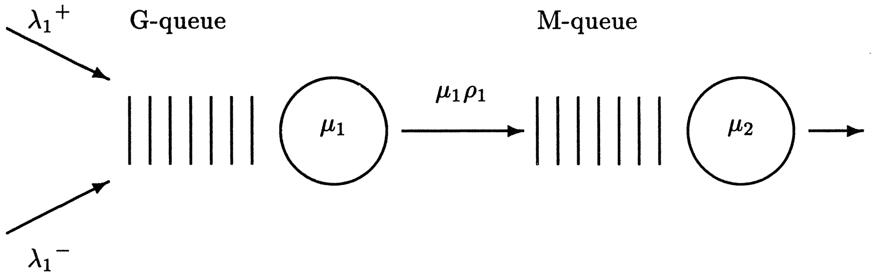
$$y_1(s) = \frac{\lambda_1{}^+}{\mu_1 + s}$$

Combining these results in equation 3 then gives

$$W^*(s) = \frac{\mu_1 - \lambda_1{}^+}{s + \mu_1 - \lambda_1{}^+}\,\frac{\mu_2 - \lambda_1{}^+}{s + \mu_2 - \lambda_1{}^+} \qquad \square$$

4.2 G-M independence

When the second queue in the pair is a normal M/M/1 queue, we have a G-M model with $\lambda_2{}^+ = \lambda_2{}^- = \lambda_2 = 0$.



The intuition is that what happens at queue 1 has no influence on the remaining sojourn time of a tagged customer in queue 2 which receives no negative customers.

Theorem 3 (G-M)

For a two server tandem network in which the first queue is an M/M/1 G-queue and the second queue is a normal M/M/1 queue, the sojourn times in each queue are independent and distributed as for the corresponding queues in isolation.

Proof
We claim that the time spent at node 2 does not depend on the numbers left behind at node 1 and node 2 (as there is no killing). Hence we verify that

$$G(x,y,z,s) = \frac{1}{1-y}\,\frac{1}{1-z}\,\frac{\mu_2}{\mu_2(1-x)+s}$$

by substituting into equation 3. Then from equation 2 we obtain

$$W^*(s) = (1-\rho_1)\,\frac{\mu_1}{\lambda_1{}^+}\,\frac{y_1(s)}{1-y_1(s)}\,\frac{\mu_2(1-\rho_2)}{\mu_2(1-\rho_2)+s}$$

which gives the expected result $W^*(s) = W_1^*(s)W_2^*(s)$, where $W_1^*(s)$ corresponds to an FCFS-RCE M/M/1 G-queue with parameters $(\lambda_1{}^+, \lambda_1{}^-, \mu_1)$ and $W_2^*(s)$ to a normal M/M/1 queue with parameters $(\mu_1\rho_1, 0, \mu_2)$ (see Harrison and Pitel [6] for the derivation of $W_1^*(s)$). □

Also note that an external Poisson arrival stream of rate $\lambda_2{}^+$ at the second M/M/1 queue would not destroy the independence between the two queues.

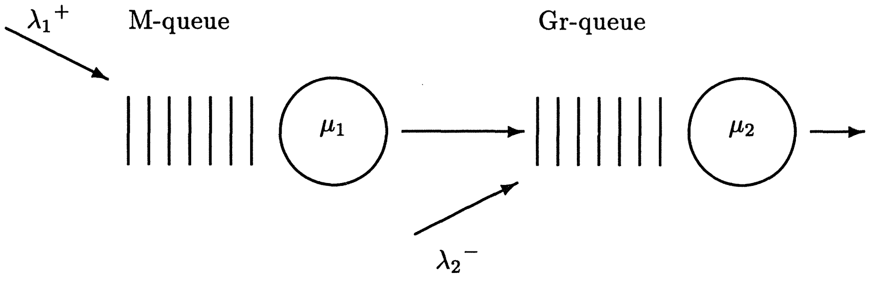
Finally, we would expect the probability that a customer is not killed in service is the probability that he is not killed in a single M/M/1 G-queue identical to the first of the tandem pair. This is confirmed by the following.

Corollary 3 (G-M)
The probability of not being killed in the G-M model is given by

$$W^*(0) = \frac{\mu_1}{\mu_1 + \lambda_1{}^-}$$

Proof
General case with $\lambda_2{}^- = 0$. □

4.3 M-Gr dependence
In this case we have a M-Gr model with $\lambda_2{}^+ = \lambda_1{}^- = 0$.



Since departures from the first queue offer protection from killing in the second queue we cannot assume that the sojourn time in the second queue is independent of the number of customers left behind on departure from the first queue. This is stated in the following.

Conjecture 1 (M-Gr)

For a two server tandem network in which the first queue is a normal M/M/1 queue and the second queue is an M/M/1 Gr-queue, the sojourn times in each queue are dependent.

Proof

Theorem 1 (G-G) simplifies to

$$W^*(s) = (1-\rho_1)(1-\rho_2)\frac{\mu_1}{\mu_1+s}\,G(\rho_2, 0, \frac{{\lambda_1}^+}{\mu_1+s}, s)$$

as $y_1 = \frac{{\lambda_1}^+}{s+\mu_1}$ and the equation for G to be solved is (after reduction of equation 3)

$$R\,G = (\mu_1 - \frac{{\lambda_1}^+}{z})\,G(x,y,0,s) - \frac{\mu_1 z}{y}\,G(x,0,z,s) + \frac{\mu_2}{(1-y)(1-z)}$$

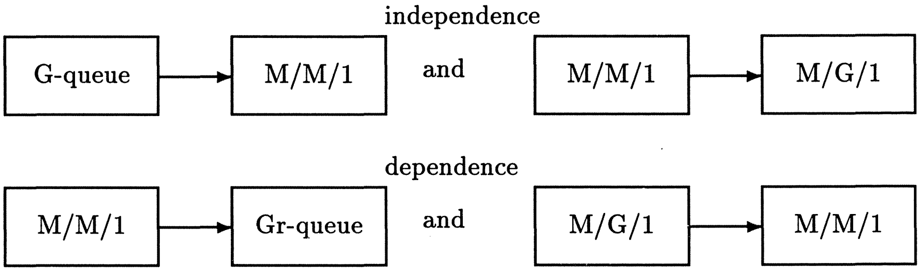
with $R(x,y,z,s) = s + {\lambda_1}^+ + {\lambda_2}^- + \mu_1 + \mu_2(1-x) - \frac{{\lambda_1}^+}{z} - {\lambda_2}^- y - \mu_1 \frac{z}{y}$. If the sojourn times were independent then G would have to be of the form

$$G(x,y,z,s) = \frac{1}{1-z}\,\frac{\mu_2}{{\lambda_2}^-}\,\frac{1}{(1-y)(1-y_1)(y_2-y)}$$

where ${\lambda_2}^- y_1 y_2 = {\lambda_1}^+$ and ${\lambda_2}^-(y_1+y_2) = s + {\lambda_2}^- + {\lambda_1}^+ + \mu_2(1-x)$ (cf. Harrison and Pitel [6]). But this proves to be wrong. So we can strongly expect dependence in this model. □

The last two sections illustrate the subtle difference between standard overtaking and our sort of overtaking. It is the reason why there is actually no inconsistency with the G-M model having the independence property but the PS-M model not (see Coffman et al. [1]); PS discipline allows overtaking in its most general sense whereas in a G-queue the influence of later customers is restricted. Both dual models – M-G and M-PS (see Mitrani [7]) respectively – do not have the independence property either because neither network is influence-free.

Finally note that M/G/1-M and its dual M-M/G/1 exhibit dependence ($G \neq M$) and independence respectively, conversely to our result for the pairs (with negative customers) G-M and its dual M-G. In the former case the dependence arises from the service time distribution whereas in ours it comes from the presence of negative customers.



The observation highlights that the casual use of the term ‘overtaking’ instead of ‘influence’ should be practised cautiously.

Similarly to the previous case, the probability of not being killed is the same as in an M/M/1 G-queue identical to the second in the tandem pair.

Corollary 4 (M-Gr)

The probability of not being killed in the M-G model is given by

$$W^*(0) = \frac{\mu_2}{\mu_2 + \lambda_2^-}$$

Proof

General case with $\lambda_1^- = 0$. □

5 CONCLUSIONS

We have seen that, even in the very simplest of G-networks - a tandem pair - the derivation of the probability distribution of response time (end-to-end delay) is difficult. This is because, in general, such networks are not overtake-free in the sense that later arriving positive customers can influence a tagged customer by providing him protection from potential killing by a future negative customer. The resulting solution requires techniques from complex analysis to solve a Riemann-Hilbert-Carleman boundary problem of the type derived by Fayolle et al. [3] to obtain the corresponding distribution in a 3-server classical network with (literal) overtaking. Indeed, our problem bears some resemblance to theirs although we adopted a somewhat different method of solution.

Although we have neither implemented our solution to provide the required Laplace transform numerically, nor (of course) a numerical inversion program, such numerical software is standard and we have detailed every step in a numerical algorithm. Moreover, we have carefully studied the dependence of the sojourn times in each queue. We found that, when the second queue is a classical M/M/1 queue, the sojourn times are independent and so the Laplace transform of response time density is the product of that for a single M/M/1 G-queue (see Harrison and Pitel [6]) and that for a classical M/M/1 queue. In all other cases, a boundary value problem must be solved and we conjecture no independence. Of course, a numerical solution could verify this conjecture, but only up to the assumption that the implementation was correct!

Although our approach is clearly extensible to tandem networks of more than two queues, the complexity of the intermediate results – all numerical solutions of Fredholm integral equations – would appear to make any exact solution totally intractable. Approximate methods would therefore be necessary, the particular one depending on the application of the model. The only obvious, practicable extension would be to a simple branching network with a single root-node and any number of possible second nodes, chosen with fixed probabilities.

REFERENCES

1. Coffman, E.G. Jnr., Fayolle, G. and Mitrani, I. (1986) Sojourn times in a tandem queue with overtaking: Reduction to a boundary value problem. *Commun. Statistics - Stochastic Models* **2(1)**, 43-65.
2. Dubner, H. and Abate, J. (1968) Numerical inversion of Laplace transforms by relating them to the finite Fourier Cosine transform. *JACM* **15(10)**, 115-123.
3. Fayolle, G., Iasnogorodski, R. and Mitrani, I. (1983) The distribution of sojourn times in a queueing network with overtaking. *Performance '83*, Agrawala, A.K. and Tripathi, S.K. (eds), North Holland, 477-486.
4. Gakhov, F.D. (1966) *Boundary value problems.* Pergamon Press.
5. Gelenbe, E. (1991) Product form networks with negative and positive customers. *J. Appl. Prob.* **28**, 656-663.
6. Harrison, P.G. and Pitel, E. (1993) Sojourn times in single-server queues with negative customers. *Journal of Applied Probability* **30**, 943–963.
7. Mitrani, I. (1985) Response time problems in communication networks. *J. Royal Statist. Soc.* **47**, 396-406.
8. Pitel, E. (1994) *Queues with negative customers.* Phd Thesis in preparation, Computing Dept., Imperial College, London.
9. Reich, E. (1957) Waiting times when queues are in tandem. *Annals of Mathematical Statistics* **28**, 768–773.

APPENDIX

Numerical values of the parameters involved:
${\lambda_1}^+ = 3, {\lambda_1}^- = 2, {\lambda_2}^+ = 0, {\lambda_2}^- = 2, \mu_1 = 2, \mu_2 = 1, s = 5$,
which gives $\rho_1 = 0.75, \rho_2 = 0.5$.

The following results are obtained with Mathematica:
$y_1 = 0, a_1 = 0, y_2 = 0.134434, y_3 = 4.544393, a_2 = 7.25, y_4 = 9.821171$,
$y_0 = 0.173344$ (C is determined by equation 19),
$y_1[s] = 0.303228$ is inside L (therefore G is evaluated with equation 16) and
L and L_e stretch between $z[y_1] = 0$ and $z[y_2] = 0.421610$, $z[y_3] = 1.108808$ and $z[y_4] = -1.166783$ respectively.

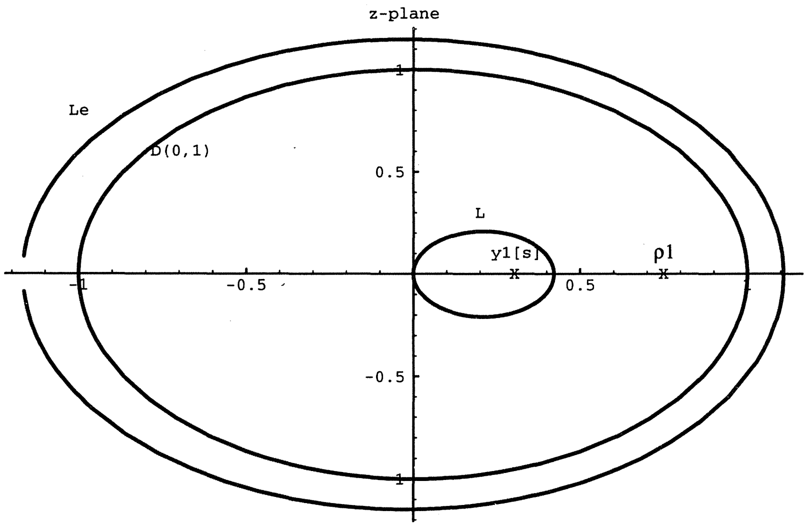


FIGURE 1. Example of a case where $\rho_1 > \rho_2$.

On the Power Series Algorithm

Ger Koole *
CWI
P.O. Box 94079
1090 GB Amsterdam, The Netherlands

The power series algorithm is a numerical procedure for solving general Markov processes. This paper gives a practical introduction to the algorithm, and presents some new results. We start by showing how the algorithm can be applied to a specific problem, the fork-join queue. Then we prove that the power series algorithm can indeed be used to solve general Markov processes. In the subsequent section we deal with the convergence properties of the algorithm. It behaves particularly well when applied to finite state processes, which is illustrated with the analysis of a bounded Petri net. We end with discussing the literature.

1 INTRODUCTION

Analytically obtaining performance measures of multi-dimensional queueing systems is often very difficult. Explicit solutions are only available for some very special models, like Jackson networks. Some specific two-dimensional models can also be solved analytically, for example by showing that solving the problem is equivalent to solving a well-studied complex analysis problem. See Boxma et al. [16] for an overview. The drawbacks of the analytical methods can be summarized as follows: the resulting problems are non-trivial to solve, we are confined to two dimensions, and small changes in the model usually lead to analytically intractable models.

On the other hand, simply numerically solving the steady state equations usually does not work well either. Often the state space is countable, giving need to truncate the state space at an appropriate level. These truncated state spaces are very big, especially if the dimension grows large and if the accuracy must be high, resulting in very long running times. Thus there is a need for efficient numerical methods to solve large Markov processes. The power series algorithm (psa) aims to be such a method. It was first developed by Hooghiemstra et al. [17] for a model in which several queues share the same servers, and later applied to several other queueing models in a series of papers by Blanc and co-authors (discussed in section 5). The examples in the next sections show that the psa often works very well, although there are some theoretical gaps.

---

*Supported by the European Grant BRA-QMIPS of CEC DG XIII

In section 2 we introduce the psa by applying the method to a fork-join queue, for which the algorithm works remarkably well. In section 3 it is shown that the psa can (formally) be applied to any Markov process with a single recurrent class. Formally, as there is no guarantee that the obtained power series converge. However, in section 4 we study the ϵ-algorithm which is capable of finding a limit for a divergent series based on its partial sums. It is shown that this works especially well for finite state processes. To illustrate this, the psa with the ϵ-algorithm is then applied to a bounded Petri net. The paper ends with a section discussing the literature.

2 THE FORK-JOIN QUEUE

In this section we give a practical introduction to the power series algorithm by applying it to a simple model with distributed processing, the *fork-join queue*. The fork-join queue consists of two or more parallel queues. Jobs arrive at the system according to a Poisson process with rate λ, and on arrival they place exactly one task in each queue (the fork primitive). This makes the arrival processes of the different queues dependent. The processing of the tasks in each queue however is independent of tasks in other queues, and is exponential with rate μ_i at queue i. Thus the marginal behaviour of each queue is that of a simple $M|M|1$ queue. A job leaves the system if all its tasks have finished service (the join primitive). As performance measure one usually takes the number of jobs in the system, which is equal to the maximum of the queue lengths (assuming that each queue services its tasks in FIFO order). This measure clearly depends on the simultaneous queue length distribution, giving need for methods to calculate it.

Many papers study the fork-join model, by using analytical methods, or by developing approximations or bounds. See section 3.1 of [16] for an overview. In this section we use the power series algorithm to compute the steady state distribution.

We use the following notation.

m - the number of queues
$x = (x_1, \ldots, x_m)$ - the state of the system
$|x| = x_1 + \ldots + x_m$
p_x - the stationary probability of state x
(A) - the indicator function of the event A
$e = (1, \ldots, 1)$
$e_i = (0, \ldots, 0, 1, 0, \ldots, 0)$ with the 1 in ith position

First take $\rho = \lambda^{1/m}$. (The choice of ρ will be motivated in the next section.) We try to write the stationary distribution as a power series of ρ. Thus we write p_x as

$$p_x = \sum_{k=0}^{\infty} a_{kx} \rho^k,$$

with a_{kx} the fixed coefficients of the power series. For the moment, assume that these power series converge. In the next section we show that $a_{kx} = 0$ if $k < |x|$. Therefore we prefer to write

$$p_x = \sum_{k=0}^{\infty} b_{kx} \rho^{|x|+k}. \tag{1}$$

The steady state equations are

$$\{\rho^m + \sum_{i=1}^{m} (x_i > 0)\mu_i\} p_x = (x > 0)\rho^m p_{x-e} + \sum_{i=1}^{m} \mu_i p_{x+e_i}. \tag{2}$$

Together with the normalizing equation

$$\sum_x p_x = 1, \tag{3}$$

they uniquely determine the steady state probabilities (assuming that the system is stable, i.e., that $\lambda = \rho^m < \min_i \mu_i$).

Now we insert (1) in (2) and (3). Equating the terms with $\rho^{|x|+k}$ in (2) gives

$$(k \geq m) b_{k-m,x} + \sum_{i=1}^{m} (x_i > 0)\mu_i b_{kx} = (x > 0) b_{k,x-e} +$$

$$(k \geq 1) \sum_{i=1}^{m} \mu_i b_{k-1,x+e_i}.$$

From (3) we get

$$b_{0,(0,0)} = 1; \quad \sum_{x:|x| \leq k} b_{k-|x|,x} = 0, \; k > 0.$$

The important observation is that we can compute the b_{kx} recursively. Starting with $b_{0,(0,0)} = 1$, we can compute successively the b_{0x} for all x with $|x| = 1, 2, \ldots$. (In practice, we compute these numbers up to a certain level, say until $|x| \leq K$.) Subsequently, we compute the $b_{1,(0,0)}$ using the normalizing equation, and we can compute b_{1x} for x with $|x| = 1, 2, \ldots$, etc. This way all coefficients with $k + |x| \leq K$ are computed.

Now the approximation is calculated by taking all other coefficients 0, which is equivalent with omitting all terms of order higher than K. Numerical results for this model with $m = 2$, $\mu_1 = 1$, $\mu_2 = 2$ and varying λ can be found in table 1. For each combination of K and λ the approximation of p_{00} and of $L = \sum_x \max\{x_1, x_2\} p_x$, the mean number of jobs in the system, is given. The exact values were calculated by truncating the state space at a sufficiently high level, and then solving the steady state equations iteratively. Note how well the psa performs, even for very small values of K, over the whole stability region of the model (the stability condition is here $\lambda < 1$). Only for λ close to 1, we have to take K large to get a reasonable estimate of L. On the other hand, even for $\lambda = 1$, for which the system is unstable, the approximation of p_{00} converges fast to 0. (Note that some computations were omitted in this case, indicated with "-".)

λ	actual values	$K = 5$	$K = 20$	$K = 50$	$K = 100$
.1	.8849, .1275	.8849, .1265	.8849, .1275	.8849, .1275	.8849, .1275
.5	.4578, 1.0688	.4568, .8279	.4578, 1.0678	.4578, 1.0688	.4578, 1.0688
.75	.2184, 3.0706	.2153, 1.4253	.2184, 2.9017	.2184, 3.0683	.2184, 3.0706
.9	.0849, 9.0419	.0800, 1.8425	.0849, 5.9041	.0849, 8.3959	.0849, 8.9956
1	0, ∞	$-$.0062, -	.0000, -	.0000, -	.0000, -

Table 1. Approximations of p_{00} and L for the fork-join queue with $\mu_1 = 1$, $\mu_2 = 2$

3 GENERAL MARKOV PROCESSES

In the previous section we applied the psa to the fork-join queue, showing that the coefficients of the power series of the stationary probabilities can be computed recursively, and using these to compute performance measures. In this section we show that the coefficients of these power series can be computed recursively for any Markov process, if the variable ρ of the power series is incorporated in the model in a suitable way.

Thus we start with an arbitrary Markov process with state space X (possibly countable) to which we want to apply the power series algorithm. We denote the transition rate from x to y by q_{xy} (for ease of notation we assume throughout that $q_{xx} = 0$). To use the psa, we consider an additional Markov process with the same state space X, but with the transitions replaced by $\rho^{f(x,y)}q_{xy}$. The problem is how to choose $f(x,y)$ such that the algorithm works for the additional process. By inserting $\rho = 1$ afterwards, we get the results for the original process. Of course, we can choose other values for ρ (as long as it amounts to a useful model), like we did in the previous section for the fork-join queue.

First we associate with every state x a *level* $l(x)$ ($l(x) \in \{0, 1, \ldots\}$). The idea is to write the stationary probability p_x as

$$p_x = \sum_k b_{kx} \rho^{l(x)+k}, \tag{4}$$

assuming that these series converge.

We show that the right choice for f is $f(x,y) = (l(y) - l(x))^+$, i.e., all transitions are of the form $\rho^{(l(y)-l(x))^+} q_{xy}$. Thus transitions to lower level states are not changed, but transitions to higher levels get a factor ρ for each level the next state is higher. To make the psa work we have to assume the following.

ASSUMPTION 3.1 *The states can be classified in levels* $0, 1, 2, \ldots$ *such that:*
(i) There is a single level 0 *state (denoted by* 0*).*
(ii) The states within each level can be ordered such that there are no transitions to higher ordered states within that level.
(iii) $\sum_{y:l(y)\leq l(x)} q_{xy} > 0$ *for all* $x \in X$*, i.e., transitions to lower level states are possible in each state.*

Because state 0 can be reached from every other state in a finite number of steps it follows directly that there can only be a single recurrent class. This is the only restriction implicated by assumption 3.1: at the end of the section we will see that we can order the states of any Markov process with a single recurrent class such that the assumption is satisfied. The choice of the ordering however can have important implications for the speed of the algorithm.

Note that the assumption implies a partial ordering of the states: $x \prec y$ if $l(x) < l(y)$ or if $l(x) = l(y)$ and there is no transition from x to y. We assume that the states are numbered $0, 1, 2, \ldots$ such that if $x \prec y$, then $x < y$.

As an illustration, consider the fork-join queue of the previous section. There we took $l(x) = |x|$, and indeed, an arrival (which increases the level by m) has a factor ρ^m. Moreover, assumption 3.1 is satisfied: the empty state is the single level 0 state, there are no transitions within each level, and in each state (except 0) there is at least one non-empty queue, making transitions to lower level states possible.

Another possible choice for l in the fork-join model is $l(x) = \max_i x_i$. Then every arrival just has a factor ρ. Now there are transitions possible within each level, but it is easily seen that they can be ordered as required in assumption 3.1(ii). Note that if the maximum is attained by two or more queues there are no transitions to strictly lower levels, but only within the same (and to higher) levels. However, that is all that is required by assumption 3.1(iii). It is easily seen that both choices of $l(x)$ basically lead to the same approximations.

The next theorem states that we can indeed write p_x in the form (4), that is, that $p_x = O(\rho^{l(x)})$.

THEOREM 3.2 *Under assumption 3.1, $p_x = O(\rho^{l(x)})$.*

Proof. We are going to use the idea of the equivalent proof given in [18] for the $BMAP|PH|1$ queue studied there. We use induction, first considering p_0. Because $p_0 = 1$ if $\rho = 0$ it is clear that $p_0 = O(1)$. Define $L_z = \{x | x \leq z\}$. Assume that $p_x = O(\rho^{l(x)})$ for all $x \in L_z$. We complete the induction step by looking at the balance equation between states in L_z and states in $X \backslash L_z$:

$$\sum_{x \in L_z} \sum_{y \notin L_z} \rho^{l(y)-l(x)} q_{xy} p_x = \sum_{x \notin L_z} \sum_{y \in L_z} q_{xy} p_x.$$

Now we show that $z' = z + 1$ is of the required order. Using the induction hypothesis, and the structure of the transitions, it is clear that the left hand side of the equation is of order $O(\rho^{l(z')})$. For z' we have that $\sum_{y \in L_z} q_{z'y} > 0$, and thus (using that all coefficients are non-negative) $p_{z'} = O(\rho^{l(z')})$. □

Next we derive the equations for which we can recursively compute the b_{kx}. The equilibrium equations are:

$$\sum_y \rho^{(l(y)-l(x))^+} q_{xy} p_x = \sum_y \rho^{(l(x)-l(y))^+} q_{yx} p_y.$$

Inserting $p_x = \rho^{l(x)} \sum_k \rho^k b_{kx}$ gives:

$$\sum_y \rho^{(l(y)-l(x))^+} q_{xy} \rho^{l(x)} \sum_k \rho^k b_{kx} = \sum_y \rho^{(l(x)-l(y))^+} q_{yx} \rho^{l(y)} \sum_k \rho^k b_{ky}.$$

Consider for fixed x the terms with $\rho^{l(x)+k}$:

$$\sum_{y:l(y)\leq l(x)} q_{xy} b_{kx} + \sum_{y:l(y)>l(x)} q_{xy} b_{k-l(y)+l(x),x} =$$

$$\sum_{y:l(y)\leq l(x)} q_{yx} b_{ky} + \sum_{y:l(y)>l(x)} q_{yx} b_{k-l(y)+l(x),y}. \tag{5}$$

From this equation we can derive b_{kx}, for $x \neq 0$, assuming we have already calculated b_{ky} for $y < x$ and b_{ly} for $l < k$ and sufficiently many y (depending on the model at hand). This can only be done if the coefficient of b_{kx} is positive (which is guaranteed by assumption 3.1(iii)) and if $q_{yx} = 0$ if $y > x$ and $l(y) = l(x)$ (guaranteed by assumption 3.1(ii)). This procedure can be repeated until all coefficients which are needed have been calculated.

The b_{k0} can be determined from $\sum_x p_x = 1$: it easily follows that $b_{00} = 1$ (if $\rho = 0$ this is the only recurrent state, by assumption 3.1(i)) and that for $k > 0$

$$\sum_{x:l(x)\leq k} b_{k-l(x),x} = 0.$$

If there is more than one level 0 state (which is the case with multiple recurrent classes) each recurrent class should be handled separately.

An important aspect of the method is the choice of l. We already saw that for specific models l could be chosen in some smart way. However, to show that the psa is applicable to general Markov processes with a single recurrent class we have to specify a choice of l which always satisfies assumption 3.1. To do so, number the states 0,1,2,..., such that from state n there is a transition possible to one or more states in $\{0, 1, \ldots, n-1\}$. This can be done easily, as long as state 0 is taken to be recurrent. Now take $l(x) = x$ for each state, and assumption 3.1 is satisfied.

So far, we have only talked about continuous time Markov processes, and not about discrete time Markov chains. They can be solved as well, simply by taking $\rho = 1$ in a model with $\sum_y q_{xy} = 1$ for each x. The only complication is that we cannot assume $q_{xx} = 0$. However, it is readily seen that the term $q_{xx} b_{kx}$ cancels on both sides of (5).

3.1 Examples

In this subsection we discuss the choice of l for several well known queueing models. Note however that for the applicability of the method the actual transition rate is not important, but just whether it is positive or not. This gives the possibility to change the models considerably without choosing another l.

Birth-death processes. Let us first consider a class of models which comprise the models discussed so far, the *m-dimensional birth-death processes.* Such a process can be seen as consisting of m queues, where arrival and departure rates depend on the state, and can occur in batches (in different queues simultaneously), but in which no arrivals and departures can occur simultaneously, avoiding transitions between queues. Thus the possible transitions out of x are of the form $x \to x+y$ or $x \to x-y$, with $y \geq 0$ (and $x-y \geq 0$). We assume that for each $x \neq 0$ there is a $y \neq 0$ such that $q_{x,x-y} > 0$. If we take $l(x) = |x|$, it is easily seen that these m-dimensional birth-death processes satisfy assumption 3.1. It is also a rich class. Not only the examples of section 2, the fork-join queue and the shortest queue model fall into it, but also the model of [5, 17] and numerous other models, like single server queues with batch arrivals, belong to it. Note that in some cases the levels can be chosen more economically, as we saw for the fork-join model.

Networks of queues. A tandem of queues is an example of a model which does not fall in the class of problems described above, but where we can take $l(x) = |x|$. Indeed, if customers enter queue 1, and join after service queue 2, ..., up to m, then the possible transitions within each level are all of the form $x \to x - e_i + e_{i+1}$, giving an ordering within level k: $(k, 0, \ldots, 0) \prec \ldots \prec (0, \ldots, 0, k)$.

For models with a more general routing structure, as in Jackson networks, this does not work any more; cycles within a level become possible. A solution is to take as state space $(x_1 + \ldots + x_m, x_2 + \ldots + x_m, \ldots, x_m)$, or, equivalently, to take $l(x) = x_1 + 2x_2 + \ldots + mx_m$. For this choice of l we can allow transitions from one queue to another, i.e., transitions of the form $x \to x - e_i + e_j$ (for x with $x_i > 0$), in addition to the batch arrivals and departures from the m-dimensional birth-death process.

Another approach, which does not fit into the framework of this section, is when we take again $l(x) = |x|$, but a transition of the form $x \to x - e_i + e_j$ with $i > j$ gets a factor ρ. Thus we have given a transition from queue i to queue j a factor ρ, although the states lie within the same level. For the other transitions the factors are taken normally, including the transitions from queue i to j if $i < j$. The psa works again in this case, and the ordering within a level is the same as for the tandem model.

Now we study models where the state of the system is not completely described by the queue lengths only: we consider polling models, where the position of the server belongs to the state, and models with an additional Markov process representing the environment (generalizing the arrival or service processes).

Markov arrival processes. First consider a single queue with arrivals according to a Markov arrival process (MAP). Assume that the states y of the MAP are numbered, such that the psa can be applied to the Markov process underlying the MAP, with levels $l(y) = y$ (which gives the restriction that there must be a single recurrent class). The states are of the form (x, y), with y the state of the MAP and x the number of customers in the queue. State (x, y) has level $x + y$. The only possible transitions within a level are of the form

$(x, y) \to (x + 1, y - 1)$, thus assumption 3.1(ii) is easily satisfied. The same holds for assumption 3.1(iii), assuring that the psa works for this model. Note that the rates at which arrivals occur do not necessarily have a factor ρ in it (as in the transition above), because the state of the MAP changes also. Only if the state of the MAP remains the same at arrival instants (the special case of a Markov modulated Poisson process), then each arrival has factor ρ. The term MAP is somewhat misleading, as it suggests that the transition rates within the Markov process governing the arrivals must be independent of x. As this is not true, it is perhaps better to speak of an auxiliary Markov process.

General service times. Such an auxiliary Markov process (AMP) can also be used to model (potential) departures from a queue with Poisson arrivals. However, when modeling departures, it is more natural to freeze the AMP (i.e., keep it in the same state until a customer arrives) when the queue is empty, instead of letting it make transitions without having customers in the queue to serve. But, as transitions to lower level states must be possible from each state except 0, it can only be frozen in state 0, which is therefore of the form $(0, 0)$. Thus the AMP can only be frozen if the transition in the AMP generating the departure is of the form $y \to 0$. If we want to be able to freeze the AMP in different states, a less obvious choice of levels has to be made.

Polling models. An interesting generalization of an auxiliary Markov process governing departures (and possibly also arrivals) is to multiple queues. As Blanc [10] shows, an important class of models which can then be modeled are the polling models. In its simplest form, the state of the AMP denotes the position of the server (i.e., at or between which queues the server is), but generalizations in different directions are possible, like the AMP denoting the service phase, or even the number of customers already served at the current queue, to be able to model for example the limited service discipline. When the server in a polling system finds an empty queue, the server usually moves to the next server; therefore the problem with freezing the departure process occurs only in the single queue case.

3.2 Memory management

At first sight it seems that what we gain by faster computations is lost again by the extra memory use, because we increased the dimension with 1. Although we need some more memory than simple iterative methods, this is usually not true. If we want to compute the stationary distribution for fixed ρ or if we want to compute a function of this distribution (like the moments of the stationary number of customers) for varying ρ, we need not keep all b_{kx} in memory.

We approximate the stationary distribution using all terms with coefficients ρ^k, with $k \leq K$. Let $N = \#\{x | l(x) \leq K\}$, the number of states with level equal to or smaller than K. Assume that the maximum number of levels a transition can go up is $\overline{k}$. It is best to compute the coefficients b_{kx} for constant value of $k + l(x)$ together. (This has the advantage that the partial sums, needed for the normalizing equation, can directly be computed and need not be kept in memory separately.) It is clear from (5) that to compute the b_{kx} with $k + l(x) = n$, only the coefficients $b_{k'x'}$ with $k' + l(x') = n - \overline{k}, \ldots, n-1$ need to be kept in memory. As soon as a coefficient is computed, it is used to update the partial sums approximating p_x. For example, in the fork-join queue only two levels are needed, while in the shortest queue model even 1 level is sufficient. In total, $N(\overline{k}+2)$ numbers need to be kept in memory: $N\overline{k}$ for the already computed coefficients, N for the coefficients now to be computed, and N for the approximations of the stationary probabilities. (Note that if $\overline{k} > K$, all coefficients need to be kept in memory.)

Now consider the situation that we want to compute a certain performance measure $\mathbb{E}f = \sum_x f(x)p_x$, say for various values of ρ. Instead of having approximations for each p_x, we keep in memory the coefficients of the power series expansion of $\mathbb{E}f$. If a coefficient b_{kx} is computed, it is added to the $(k + l(x))$th coefficient of the expansion of $\mathbb{E}f$. Now, $N(\overline{k}+1) + K$ numbers are necessary. Note that the same performance measure can be computed for several ρ. Using this, the coefficients b_{kx} need only to be computed once to supply the results produced by the psa for the examples of section 2. If we are only interested in $\mathbb{E}f$ for a specific value of ρ, then $N(\overline{k}+1)+1$ numbers are sufficient.

4 CONVERGENCE

In section 2 we saw that the power series expansions of p_{00} and L in the fork-join queue converge for all values of ρ for which the system is stable. However, in general this need not be the case. This is not surprising: the psa develops each stationary probability as a power series around $\rho = 0$, and the radius of convergence of such a series is in general unknown. This section is devoted to the study of the convergence properties.

To illustrate the problems, we consider the *shortest queue model* in section 4.1, and show that the power series involved do not converge for certain values of ρ. To improve the convergence properties we make use of an algorithm applicable to arbitrary power series, the *ϵ-algorithm*, which was first used by Blanc in conjunction with the psa. The ϵ-algorithm is the subject of section 4.2.

In section 4.3 we take a closer look at finite state Markov processes. The ϵ-algorithm is particularly well suited to deal with this type of processes. Finally in section 4.4 we use the results to analyze a bounded Petri net.

4.1 The shortest queue model

In the shortest queue model, we have a Poisson(λ) stream of jobs arriving at m parallel queues. Each job consists of a single task which is routed to the shortest queue. In case of a tie, each queue is selected with equal probability. The server at queue i again has service rate μ_i. Simply choosing $\rho = \lambda$ makes the psa work here. We will not go into all details of the balance equations, but we just give the equation from which the b_{kx} are derived (for $m = 2$):

$$(k > 0)b_{k-1,x} + \sum_{i=1,2} (x_i > 0)\mu_i b_{kx} =$$

$$\Big((x_1 > 0)(x_1 \leq x_2) + \tfrac{1}{2}(x_1 = x_2 + 1)\Big) b_{k,x-e_1} +$$

$$\Big((x_2 > 0)(x_2 \leq x_1) + \tfrac{1}{2}(x_2 = x_1 + 1)\Big) b_{k,x-e_2} +$$

$$(k > 0) \sum_{i=1,2} \mu_i b_{k-1,x+e_i}.$$

The numerical results for $\mu_1 = 1$, $\mu_2 = 2$ and $K = 50$ can be found in table 2. For different values of λ, p_{00} and $L = \sum_x (x_1 + x_2) p_x$, the mean number of jobs in the system, are computed. Clearly, the power series expansions for $\lambda = 1.5$ do not converge; in this case the power series expansion of the p_x converges only if $\lambda \leq 1$. Note that the stability condition here is $\lambda < \mu_1 + \mu_2$, thus the psa does not always give satisfactory answers in the whole stability region. Elaborate numerical experiments with this model can be found in [4, 11].

λ	actual values	$K = 50$
.1	.9279, .0746	.9279, .0746
.5	.6859, .3824	.6859, .3825
1	.4551, .9523	.4550, .8526
1.5	.2806, 1.5401	−29.5875, 30.8450

Table 2. Approximations of p_{00} and L for the shortest queue model with $\mu_1 = 1$, $\mu_2 = 2$

4.2 The ϵ-algorithm

The ϵ-algorithm was introduced by Wynn (see e.g. [20]) to accelerate the convergence of power series. Given the partial sums $S_m = \sum_{k=0}^{m} c_k \rho^k$, a two-dimensional array with entries $\epsilon_r^{(m)}$ is computed, using the formula

$$\epsilon_{r+1}^{(m)} = \epsilon_{r-1}^{(m+1)} + (\epsilon_r^{(m+1)} - \epsilon_r^{(m)})^{-1},$$

with initial conditions

$$\epsilon_{-1}^{(m)} = 0, \ m = 1, 2, \ldots,$$

and

$$\epsilon_0^{(m)} = S_m, \ m = 0, 1, \ldots.$$

Now $\epsilon_r^{(m)}$ with r even is used instead of S_m to approximate the limit S_∞. The numbers $\epsilon_r^{(m)}$ with r odd are only used as intermediate results.

The idea behind the ϵ-algorithm is that $\epsilon_{2r}^{(m)}$ approximates S_∞ by a quotient of polynomials, the numerator of degree $m + r$, the denominator of degree r, which are completely determined by the first $2r + m$ coefficients of the power series to be approximated. In the cases considered in this paper, the zeros of the denominator apparently converge to the singularities of S_∞, thereby extending the region of convergence.

Although the ϵ-algorithm involves repeated subtraction and division, Wynn [20] states that it is often remarkably stable. This is in compliance with our findings.

We applied the psa with the ϵ-algorithm to the shortest queue model of section 2. The results can be found in table 3. We see that the psa together with the ϵ-algorithm gives the correct answers for all values of λ.

λ	actual values	$K = 50$
.1	.9279, .0746	.9279, .0746
.5	.6859, .3824	.6859, .3825
1	.4551, .9523	.4550, .8526
1.5	.2806, 1.5401	.2805, 1.5415
2	.1509, 2.7628	.1506, 2.7751
3	0, ∞	.0000, 2.28×10^6

Table 3. Approximations of p_{00} and L for the shortest queue model with $\mu_1 = 1$, $\mu_2 = 2$, using the ϵ-algorithm

For the models we studied the ϵ-algorithm works very well. However, if $\epsilon_r^{(m+1)} = \epsilon_r^{(m)}$ for some values of r and m, problems may arise, because $\epsilon_{r+1}^{(m)}$ cannot be computed any more. A simple example is $(1-\rho^3)^{-1}$, for which the power series expansion has coefficients $1, 0, 0, 1, 0, 0, \ldots$. Application of the ϵ-algorithm easily shows what goes wrong. On the other hand, sometimes we find $\epsilon_{2r}^{(m+1)} = \epsilon_{2r}^{(m)}$ for all $m \geq m'$. Then $\epsilon_{2r}^{(m')}$ is exactly $\sum_{k=0}^{\infty} c_k \rho^k$. Some special cases for which this occurs are identified in the next subsection.

Remark. Apart from the ϵ-algorithm, a method using conformal mappings can be applied to improve the convergence properties. It is based on putting $\rho = \theta/(1 + G - G\theta)$ (with G a positive constant), and then calculating p_x as a power series in θ. Using the notation

$$p_x = \sum_{k=0}^{\infty} u_{kx} \theta^{l(x)+k},$$

it can be shown that the u_{kx} can be computed recursively if the b_{kx} can be computed recursively. Moreover, all singularities of the expansion of p_x inside the unit disk (causing the power series not to converge for $\rho = 1$) but outside the disk with center $\rho = 1/2$ and radius $1/2$, can be removed with such a conformal mapping. Thus, the psa can again be applied, and the convergence properties are considerably improved. For more details, see for example [5].

4.3 Finite State Processes

In this section we study Markov processes which have a finite state space, with N elements. This will allow us to write the stationary probabilities as quotients of polynomials in ρ. From this we conclude that the ϵ-algorithm, if applicable, produces exact results.

Let G be the infinitesimal generator of the Markov process, i.e., $g_{ij} = q_{ij}$ if $i \neq j$, and $g_{ii} = -\sum_j q_{ij}$. Construct G' from G by replacing the last column by e. Then the steady state vector is the unique solution of the equation $pG' = e_N$, if we assume that the process consists of a single recurrent class. Note that all elements of G' are polynomials of ρ.

To compute the stationary probabilities p_x we can apply Cramer's rule, that is,

$$p_x = \frac{|G'_x|}{|G'|},$$

where G'_x is obtained from G' by replacing the xth row by e_N, and where we denote by $|\cdot|$ the determinant of a matrix. As all entries of both matrices are polynomials in ρ, we conclude that p_x is a quotient of polynomials in ρ, i.e., it is a rational function.

Again, assume that the maximum number of levels a transition can go up is $\overline{k}$. Then all entries are of order $\leq \overline{k}$, and as the last column consists of 1's, both determinants are of order $\leq (N-1)\overline{k}$. As it is useless to have more levels than states, and thus $\overline{k} \leq N-1$, we can assume in general that each determinant is of order $\leq (N-1)^2$.

From [20] we know that $\epsilon_{2r}^{(0)}$ approximates S_∞ with a uniquely determined rational function where both the numerator and the denominator are of order r. Thus to compute the stationary probabilities exactly it is sufficient to compute $\epsilon_{2(N-1)\overline{k}}^{(0)}$. If $\overline{k}$ is small (in most examples we had $\overline{k} = 1$), this can often be done, even for reasonably sized models.

Another interesting implication of p_x being a rational function is that p_x is analytic in $\rho = 0$. Indeed, p_x has a finite number of poles, each of which is unequal to 0, as $p_x = (x = 0)$ for $\rho = 0$. Thus, for ρ small enough, the power series converge, without applying the ϵ-algorithm. If ρ is the traffic intensity (as it was in most queueing examples), this leads to a light traffic result.

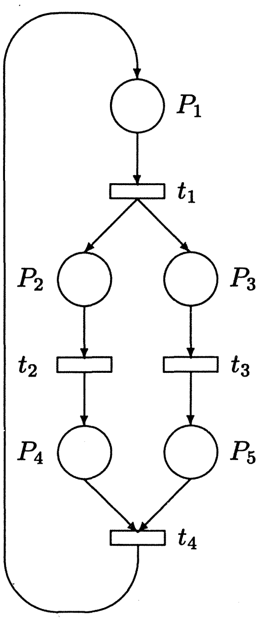


Figure 1. A stochastic Petri net

4.4 Petri Net Example

To illustrate the ideas from the previous subsection, we analyze the simple stochastic Petri net depicted in figure 1. We denote its markings with $(x_1, \dots, x_5)$, where x_i is the number of tokens at place P_i. As initial markings we take $(n, 0, 0, 0, 0)$, for various n. This marked graph is live and bounded, and to represent its reachability set we can restrict ourselves to (x_1, x_2, x_3), as $x_4 = n - x_1 - x_2$ and $x_5 = n - x_1 - x_3$. Transition t_i has an exponential firing time with rate λ_i.

Note that this Petri net is strongly related to the fork-join queue. Indeed, transition t_1 corresponds to the fork primitive. Transitions t_2 and t_3 correspond to the distributed processing of the tasks, and transition t_4 is only enabled if there is both a token at P_4 and at P_5, that is, if a job has finished service in the fork-join queue. Thus, in queueing terms, the Petri net exists of a closed cycle of three centers, one of which is a fork-join queue, and two of them are simple single server queues.

To apply the psa, we have to partition the state space into levels. We took as levels $l(x_1, x_2, x_3) = n - x_1$. Consequently, transition t_1 gets a term ρ, thus λ_1 is replaced by $\rho\lambda_1$. Equation (5) becomes:

$$b_{kx}\{\lambda_2(x_2 > 0) + \lambda_3(x_3 > 0) + \lambda_4(x_1 + x_2 < n, x_1 + x_3 < n)\} +$$

$$b_{k-1,x}\lambda_1(x_1 > 0, k > 0) =$$

$$b_{k,(x_1+1,x_2-1,x_3-1)}\lambda_1(x_2 > 0, x_3 > 0) +$$

$$b_{k,(x_1,x_2+1,x_3)}\lambda_2(x_1 + x_2 < n) + b_{k,(x_1,x_2,x_3+1)}\lambda_3(x_1 + x_3 < n) +$$

$$b_{k-1,(x_1-1,x_2,x_3)}\lambda_4(x_1 > 0, k > 0).$$

We are interested in the throughput of the system, i.e., the average number of firings of t_1 per unit of time. This is equivalent to computing the stationary probability of having 0 tokens in P_1 (denoted by p), as the throughput is equal to $(1-p)\lambda_1$.

First we computed the coefficients of the power series of p. As $p = \sum_{x_2+x_3 \leq n} p_{(0,x_2,x_3)}$, this is the sum of the stationary probabilities of all level n states. Thus, the first n coefficients of this power series are 0. There are no transitions 2 or more levels up, and therefore only 2 arrays the size of the state space (which is $N = 1^2 + 2^2 + \ldots + (n+1)^2 = (n+1)(n+2)(2n+3)/6$) and 1 array with the coefficients of p need to be kept in memory, according to section 3.2. After computing all coefficients of p up to a certain K, we applied the ϵ-algorithm, after omitting the trailing zeros. To apply this algorithm, 3 arrays of size K have to be stored.

Typical output for $\lambda_i = 1$ and $n = 3$ (30 states) can be found in table 4, where $\epsilon_r^{(m)}$ can be found for the series without trailing zeros. For reasons of space we left out the $\epsilon_r^{(m)}$ with $m > 7$.

	$m = 0$	1	2	3	4	5	6	7
$r = 0$	6.125000	-8.593750	1.455729	0.643993	51.176851	-128.409830	99.173984	-14.747156
1	-0.067941	0.099508	-1.231927	0.019789	-0.005568	0.004394	-0.008778	0.002312
2	-2.621754	0.704660	1.442896	11.740703	-28.031678	23.255298	75.427968	63.409885
3	0.400132	0.122653	0.116897	-0.030711	0.023892	0.010389	-0.080896	-0.003497
4	-2.899217	-172.300284	4.966029	-9.717844	-50.802362	64.473330	76.329890	-84.021912
5	0.116750	0.122538	-0.098813	-0.000448	0.019064	0.003445	-0.009733	0.002105
6	0.448322	0.448332	0.448334	0.448328	0.448333	0.448330	0.448332	
7	1.03×10^5	6.14×10^5	-1.73×10^5	1.78×10^5	-2.56×10^5	4.30×10^5		
8	0.448334	0.448332	0.448331	0.448331	0.448331			
9	-1.14×10^4	-7.77×10^5	2.38×10^6	-6.91×10^6				
10	0.448331	0.448331	0.448331					
11	-2.95×10^7	3.65×10^7						
12	0.448331							

Table 4. Approximations $\epsilon_r^{(m)}$ of p for the Petri net example with $\lambda_i = 1$ and $n = 3$

The table shows some interesting phenomena. As r gets large, for r even, the approximation gets better. Note that as $\epsilon_r^{(m)}$ gets close to $\epsilon_r^{(m+1)}$ for r even, then $\epsilon_{r+1}^{(m)}$ (which is only used as intermediate step, as $r+1$ is odd) gets very large. This does not lead to numerical instabilities (at least not in this case), as can be seen from the table. Even if we take r very large, we find the correct answer: e.g., $\epsilon_{100}^{(0)} = 0.448331$.

Note that, as $\overline{k} = 1$ and the number of states is 30, $\epsilon_{58}^{(0)}$ should give the correct answer (and it does: 0.448331).

In the following table results are given for various values of n. The first column gives n, the second the total number of states N, the third the computed value of p, and the fourth the lowest value of r for which $\epsilon_r^{(0)}$ approximates p with a precision of 5 digits. Note that this value of r is considerably lower than $2(N-1)$, the value for which $\epsilon_r^{(0)} = S_\infty$. This is done again for $\lambda_i = 1$. A good indication that the approximation is close are the values of $\epsilon_r^{(m)}$ for r odd; if they are big, $\epsilon_{r+1}^{(0)}$ is close. To compute $\epsilon_r^{(0)}$ for $n = 100$ and for r up to 500 took ≈ 15 minutes on a fast workstation.

n	N	p	r
1	5	.714286	2
2	14	.551546	6
3	30	.448331	8
5	91	.325768	16
10	506	.193286	30
25	6201	.087018	82
50	45526	.045405	198
100	348551	.023207	382

Table 5. Approximations of p for the Petri net example with $\lambda_i = 1$

For $n = 1$ the computation of the b_{kx} can easily be done by hand, and we find (for general firing rates) that $p = \alpha - \alpha^2\rho + \alpha^3\rho^2 - \ldots$, with $\alpha = \lambda_1(\lambda_2 + \lambda_3 + 3\lambda_4)/((\lambda_2 + \lambda_3)\lambda_4)$. If we apply the ϵ-algorithm once, i.e., if we compute $\epsilon_2^{(m)}$, we find that $\epsilon_2^{(m)} = \alpha(1 + \alpha\rho)^{-1}$ for all m. Thus indeed, if we take $\lambda_i = \rho = 1$, we get $p = \frac{5}{7} \approx 0.714286$, coinciding with our numerical results.

5 LITERATURE

The psa was introduced in Hooghiemstra et al. [17], where it is applied to a coupled processor model, which is a special case of the m-dimensional birth-death process discussed in section 3.1. (They assume single arrivals and departures, and they have some restrictions on the summed transition rates.) For this model it is shown that the psa can be applied and that p_x is analytic in $\rho = 0$, giving a light traffic result (as ρ was taken equal to the arrival rate).

Next Blanc started to work on the psa, resulting in a series of papers with various co-authors, [2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18]. The convergence properties being the main hurdle for the application of the algorithm to various models, [5] proposes the use of conformal mappings (based on a private communication with Keane, Hooghiemstra & Van de Ree), and applies it successfully to a few models. More extensive computations on one of these models, the shortest queue model, can be found in [4]. The coupled processor model is studied again in [6].

The ϵ-algorithm is first used in [7], for a type of multi-dimensional queueing model in which the server cyclically visits all queues, serving a geometric number of customers (if available) at each queue. This is called a Bernoulli schedule. There are no switching times, resulting in a single level 0 state. Also the shortest queue model is studied again ([11]), now with the ϵ-algorithm.

In Blanc [10] a multi-dimensional queueing model is studied, where the arrivals and departures (which occur one by one) are governed by an additional state component. The transition rates are allowed to depend on the entire state of the system, and arrivals and departures occur simultaneously with state transitions of the extra component. Many polling models fall into this framework. Denote, like we did for the polling model of section 3.1, the state with (x, y). Blanc takes as levels $l(x, y) = |x|$. Because of the transitions in the AMP, the $b_{k,(x,y)}$ cannot be solved recursively, but for each level there remains a set of equations, as many as there are states in the AMP. By solving this set of equations for each level, a solution is found. (Note the difference with the method proposed for these types of models in section 3.1.) The method proposed in [10] is applied to various polling models in Blanc [8, 9, 10], Blanc & Van der Mei [13, 14, 15], Altman et al. [2] and Altman [1]. A model with a single queue and batch arrivals is solved in a similar way in Van den Hout & Blanc [18]. In [13] it is shown that the psa can also be used to compute derivatives in addition to steady state probabilities. This is used to compute optimal values for certain system parameters.

A recent and fairly complete overview of Blanc's work on the psa is [12].

Bavinck et al. [3] study the coupled processor model with 2 queues, equal arrival rates in both queues, and a server that serves both queues with rate $\mu/2$ if $x > 0$; if one queue is empty it serves the other queue with rate μ. For this model the coefficients on the diagonal, i.e., the numbers b_{kx} with $x_1 = x_2$, are computed explicitly, and error bounds are derived.

In a way closely related to the psa Levine & Finkel [19] study the optimal routing for a queueing model. They write the discounted costs under each policy as a power series of the arrival rate to the system, and derive the first few terms using the optimality equation for the system. This leads to optimal policies under low (and in a similar way, high) traffic.

Acknowledgement. I like to thank Prof. J.W. Cohen for many interesting discussions on this subject.

REFERENCES

1. E. Altman. Analysing timed-token ring protocols using the power series algorithm. In *Proceedings of the 14th International Teletraffic Conference*, 1994.
2. E. Altman, J.P.C. Blanc, A. Khamisy, and U. Yechiali. Polling systems with walking and switch-in times. To appear in *Stochastic Models*, 10, 1994.
3. H. Bavinck, G. Hooghiemstra, and E. de Waard. An application of Gegenbauer polynomials in queueing theory. *Journal of Computational and Applied Mathematics*, 49:1–10, 1993.
4. J.P.C. Blanc. A note on waiting times in systems with queues in parallel. *Journal of Applied Probability*, 24:540–546, 1987.
5. J.P.C. Blanc. On a numerical method for calculating state probabilities for queueing systems with more than one waiting line. *Journal of Computational and Applied Mathematics*, 20:119–125, 1987.

6. J.P.C. Blanc. A numerical study of a coupled processor model. In G. Iazeolla, P.J. Courtois, and O.J. Boxma, editors, *Computer Performance and Reliability*, pages 289–303. North-Holland, 1988.
7. J.P.C. Blanc. A numerical approach to cyclic-service queueing models. *Queueing Systems*, 6:173–188, 1990.
8. J.P.C. Blanc. The power-series algorithm applied to cyclic polling systems. *Stochastic Models*, 7:527–545, 1991.
9. J.P.C. Blanc. An algorithmic solution of polling models with limited service disciplines. *IEEE Transactions on Communications*, 40:1152–1155, 1992.
10. J.P.C. Blanc. Performance evaluation of polling systems by means of the power-series algorithm. *Annals of Operations Research*, 35:155–186, 1992.
11. J.P.C. Blanc. The power-series algorithm applied to the shortest-queue model. *Operations Research*, 40:157–167, 1992.
12. J.P.C. Blanc. Performance analysis and optimization with the power-series algorithm. In L. Donatiello and R. Nelson, editors, *Performance Evaluation of Computer and Communication Systems*, pages 53–80. Springer-Verlag, 1993. Lecture Notes in Computer Science 729.
13. J.P.C. Blanc and R.D. van der Mei. Optimization of polling systems by means of gradient methods and the power series algorithm. Technical Report FEW 575, Tilburg University, 1992.
14. J.P.C. Blanc and R.D. van der Mei. Optimization of polling systems with Bernoulli schedules. Technical Report FEW 563, Tilburg University, 1992.
15. J.P.C. Blanc and R.D. van der Mei. The power series algorithm applied to polling systems with a dormant server. In J. Labetoulle and J.W. Roberts, editors, *The Fundamental Role of Teletraffic in the Evolution of Telecommunications Networks*, pages 865–874. Elsevier, 1994.
16. O.J. Boxma, G.M. Koole, and Z. Liu. Queueing-theoretic solution methods for models of parallel and distributed systems. In: These proceedings, 1994.
17. G. Hooghiemstra, M. Keane, and S. van de Ree. Power series for stationary distributions of coupled processor models. *SIAM Journal on Applied Mathematics*, 48:1159–1166, 1988.
18. W.B. van den Hout and J.P.C. Blanc. The power-series algorithm extended to the $BMAP|PH|1$ queue. Submitted for publication, 1993.
19. A. Levine and D. Finkel. Load balancing in a multi-server queueing system. *Computers and Operations Research*, 17:17–25, 1990.
20. P. Wynn. On the convergence and stability of the epsilon algorithm. *SIAM Journal on Numerical Analysis*, 3:91–122, 1966.

MATHEMATICAL CENTRE TRACTS

1 T. van der Walt. *Fixed and almost fixed points.* 1963.

2 A.R. Bloemena. *Sampling from a graph.* 1964.

3 G. de Leve. *Generalized Markovian decision processes, part I: model and method.* 1964.

4 G. de Leve. *Generalized Markovian decision processes, part II: probabilistic background.* 1964.

5 G. de Leve, H.C. Tijms, P.J. Weeda. *Generalized Markovian decision processes, applications.* 1970.

6 M.A. Maurice. *Compact ordered spaces.* 1964.

7 W.R. van Zwet. *Convex transformations of random variables.* 1964.

8 J.A. Zonneveld. *Automatic numerical integration.* 1964.

9 P.C. Baayen. *Universal morphisms.* 1964.

10 E.M. de Jager. *Applications of distributions in mathematical physics.* 1964.

11 A.B. Paalman-de Miranda. *Topological semigroups.* 1964.

12 J.A.Th.M. van Berckel, H. Brandt Corstius, R.J. Mokken, A. van Wijngaarden. *Formal properties of newspaper Dutch.* 1965.

13 H.A. Lauwerier. *Asymptotic expansions.* 1966, out of print: replaced by MCT 54.

14 H.A. Lauwerier. *Calculus of variations in mathematical physics.* 1966.

15 R. Doornbos. *Slippage tests.* 1966.

16 J.W. de Bakker. *Formal definition of programming languages with an application to the definition of ALGOL 60.* 1967.

17 R.P. van de Riet. *Formula manipulation in ALGOL 60, part 1.* 1968.

18 R.P. van de Riet. *Formula manipulation in ALGOL 60, part 2.* 1968.

19 J. van der Slot. *Some properties related to compactness.* 1968.

20 P.J. van der Houwen. *Finite difference methods for solving partial differential equations.* 1968.

21 E. Wattel. *The compactness operator in set theory and topology.* 1968.

22 T.J. Dekker. *ALGOL 60 procedures in numerical algebra, part 1.* 1968.

23 T.J. Dekker, W. Hoffmann. *ALGOL 60 procedures in numerical algebra, part 2.* 1968.

24 J.W. de Bakker. *Recursive procedures.* 1971.

25 E.R. Paërl. *Representations of the Lorentz group and projective geometry.* 1969.

26 European Meeting 1968. *Selected statistical papers, part I.* 1968.

27 European Meeting 1968. *Selected statistical papers, part II.* 1968.

28 J. Oosterhoff. *Combination of one-sided statistical tests.* 1969.

29 J. Verhoeff. *Error detecting decimal codes.* 1969.

30 H. Brandt Corstius. *Exercises in computational linguistics.* 1970.

31 W. Molenaar. *Approximations to the Poisson, binomial and hypergeometric distribution functions.* 1970.

32 L. de Haan. *On regular variation and its application to the weak convergence of sample extremes.* 1970.

33 F.W. Steutel. *Preservations of infinite divisibility under mixing and related topics.* 1970.

34 I. Juhász, A. Verbeek, N.S. Kroonenberg. *Cardinal functions in topology.* 1971.

35 M.H. van Emden. *An analysis of complexity.* 1971.

36 J. Grasman. *On the birth of boundary layers.* 1971.

37 J.W. de Bakker, G.A. Blaauw, A.J.W. Duijvestijn, E.W. Dijkstra, P.J. van der Houwen, G.A.M. Kamsteeg-Kemper, F.E.J. Kruseman Aretz, W.L. van der Poel, J.P. Schaap-Kruseman, M.V. Wilkes, G. Zoutendijk. *MC-25 Informatica Symposium.* 1971.

38 W.A. Verloren van Themaat. *Automatic analysis of Dutch compound words.* 1972.

39 H. Bavinck. *Jacobi series and approximation.* 1972.

40 H.C. Tijms. *Analysis of (s,S) inventory models.* 1972.

41 A. Verbeek. *Superextensions of topological spaces.* 1972.

42 W. Vervaat. *Success epochs in Bernoulli trials (with applications in number theory).* 1972.

43 F.H. Ruymgaart. *Asymptotic theory of rank tests for independence.* 1973.

44 H. Bart. *Meromorphic operator valued functions.* 1973.

45 A.A. Balkema. *Monotone transformations and limit laws.* 1973.

46 R.P. van de Riet. *ABC ALGOL, a portable language for formula manipulation systems, part 1: the language.* 1973.

47 R.P. van de Riet. *ABC ALGOL, a portable language for formula manipulation systems, part 2: the compiler.* 1973.

48 F.E.J. Kruseman Aretz, P.J.W. ten Hagen, H.L. Oudshoorn. *An ALGOL 60 compiler in ALGOL 60, text of the MC-compiler for the EL-X8.* 1973.

49 H. Kok. *Connected orderable spaces.* 1974.

50 A. van Wijngaarden, B.J. Mailloux, J.E.L. Peck, C.H.A. Koster, M. Sintzoff, C.H. Lindsey, L.G.L.T. Meertens, R.G. Fisker (eds.). *Revised report on the algorithmic language ALGOL 68.* 1976.

51 A. Hordijk. *Dynamic programming and Markov potential theory.* 1974.

52 P.C. Baayen (ed.). *Topological structures.* 1974.

53 M.J. Faber. *Metrizability in generalized ordered spaces.* 1974.

54 H.A. Lauwerier. *Asymptotic analysis, part 1.* 1974.

55 M. Hall, Jr., J.H. van Lint (eds.). *Combinatorics, part 1: theory of designs, finite geometry and coding theory.* 1974.

56 M. Hall, Jr., J.H. van Lint (eds.). *Combinatorics, part 2: graph theory, foundations, partitions and combinatorial geometry.* 1974.

57 M. Hall, Jr., J.H. van Lint (eds.). *Combinatorics, part 3: combinatorial group theory.* 1974.

58 W. Albers. *Asymptotic expansions and the deficiency concept in statistics.* 1975.

59 J.L. Mijnheer. *Sample path properties of stable processes.* 1975.

60 F. Göbel. *Queueing models involving buffers.* 1975.

63 J.W. de Bakker (ed.). *Foundations of computer science.* 1975.

64 W.J. de Schipper. *Symmetric closed categories.* 1975.

65 J. de Vries. *Topological transformation groups, 1: a categorical approach.* 1975.

66 H.G.J. Pijls. *Logically convex algebras in spectral theory and eigenfunction expansions.* 1976.

68 P.P.N. de Groen. *Singularly perturbed differential operators of second order.* 1976.

69 J.K. Lenstra. *Sequencing by enumerative methods.* 1977.

70 W.P. de Roever, Jr. *Recursive program schemes: semantics and proof theory.* 1976.

71 J.A.E.E. van Nunen. *Contracting Markov decision processes.* 1976.

72 J.K.M. Jansen. *Simple periodic and non-periodic Lamé functions and their applications in the theory of conical waveguides.* 1977.

73 D.M.R. Leivant. *Absoluteness of intuitionistic logic.* 1979.

74 H.J.J. te Riele. *A theoretical and computational study of generalized aliquot sequences.* 1976.

75 A.E. Brouwer. *Treelike spaces and related connected topological spaces.* 1977.

76 M. Rem. *Associons and the closure statements.* 1976.

77 W.C.M. Kallenberg. *Asymptotic optimality of likelihood ratio tests in exponential families.* 1978.

78 E. de Jonge, A.C.M. van Rooij. *Introduction to Riesz spaces.* 1977.

79 M.C.A. van Zuijlen. *Empirical distributions and rank statistics.* 1977.

80 P.W. Hemker. *A numerical study of stiff two-point boundary problems.* 1977.

81 K.R. Apt, J.W. de Bakker (eds.). *Foundations of computer science II, part 1.* 1976.

82 K.R. Apt, J.W. de Bakker (eds.). *Foundations of computer science II, part 2.* 1976.

83 L.S. van Benthem Jutting. *Checking Landau's "Grundlagen" in the AUTOMATH system.* 1979.

84 H.L.L. Busard. *The translation of the elements of Euclid from the Arabic into Latin by Hermann of Carinthia (?), books vii-xii.* 1977.

85 J. van Mill. *Supercompactness and Wallmann spaces.* 1977.

86 S.G. van der Meulen, M. Veldhorst. *Torrix I, a programming system for operations on vectors and matrices over arbitrary fields and of variable size.* 1978.

88 A. Schrijver. *Matroids and linking systems.* 1977.

89 J.W. de Roever. *Complex Fourier transformation and analytic functionals with unbounded carriers.* 1978.

90 L.P.J. Groenewegen. *Characterization of optimal strategies in dynamic games.* 1981.

91 J.M. Geysel. *Transcendence in fields of positive characteristic.* 1979.

92 P.J. Weeda. *Finite generalized Markov programming.* 1979.

93 H.C. Tijms, J. Wessels (eds.). *Markov decision theory.* 1977.

94 A. Bijlsma. *Simultaneous approximations in transcendental number theory.* 1978.

95 K.M. van Hee. *Bayesian control of Markov chains.* 1978.

96 P.M.B. Vitányi. *Lindenmayer systems: structure, languages, and growth functions.* 1980.

97 A. Federgruen. *Markovian control problems; functional equations and algorithms.* 1984.

98 R. Geel. *Singular perturbations of hyperbolic type.* 1978.

99 J.K. Lenstra, A.H.G. Rinnooy Kan, P. van Emde Boas (eds.). *Interfaces between computer science and operations research.* 1978.

100 P.C. Baayen, D. van Dulst, J. Oosterhoff (eds.). *Proceedings bicentennial congress of the Wiskundig Genootschap, part 1.* 1979.

101 P.C. Baayen, D. van Dulst, J. Oosterhoff (eds.). *Proceedings bicentennial congress of the Wiskundig Genootschap, part 2.* 1979.

102 D. van Dulst. *Reflexive and superreflexive Banach spaces.* 1978.

103 K. van Harn. *Classifying infinitely divisible distributions by functional equations.* 1978.

104 J.M. van Wouwe. *GO-spaces and generalizations of metrizability.* 1979.

105 R. Helmers. *Edgeworth expansions for linear combinations of order statistics.* 1982.

106 A. Schrijver (ed.). *Packing and covering in combinatorics.* 1979.

107 C. den Heijer. *The numerical solution of nonlinear operator equations by imbedding methods.* 1979.

108 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science III, part 1.* 1979.

109 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science III, part 2.* 1979.

110 J.C. van Vliet. *ALGOL 68 transput, part I: historical review and discussion of the implementation model.* 1979.

111 J.C. van Vliet. *ALGOL 68 transput, part II: an implementation model.* 1979.

112 H.C.P. Berbee. *Random walks with stationary increments and renewal theory.* 1979.

113 T.A.B. Snijders. *Asymptotic optimality theory for testing problems with restricted alternatives.* 1979.

114 A.J.E.M. Janssen. *Application of the Wigner distribution to harmonic analysis of generalized stochastic processes.* 1979.

115 P.C. Baayen, J. van Mill (eds.). *Topological structures II, part 1.* 1979.

116 P.C. Baayen, J. van Mill (eds.). *Topological structures II, part 2.* 1979.

117 P.J.M. Kallenberg. *Branching processes with continuous state space.* 1979.

118 P. Groeneboom. *Large deviations and asymptotic efficiencies.* 1980.

119 F.J. Peters. *Sparse matrices and substructures, with a novel implementation of finite element algorithms.* 1980.

120 W.P.M. de Ruyter. *On the asymptotic analysis of large-scale ocean circulation.* 1980.

121 W.H. Haemers. *Eigenvalue techniques in design and graph theory.* 1980.

122 J.C.P. Bus. *Numerical solution of systems of nonlinear equations.* 1980.

123 I. Yuhász. *Cardinal functions in topology - ten years later.* 1980.

124 R.D. Gill. *Censoring and stochastic integrals.* 1980.

125 R. Eising. *2-D systems, an algebraic approach.* 1980.

126 G. van der Hoek. *Reduction methods in nonlinear programming.* 1980.

127 J.W. Klop. *Combinatory reduction systems.* 1980.

128 A.J.J. Talman. *Variable dimension fixed point algorithms and triangulations.* 1980.

129 G. van der Laan. *Simplicial fixed point algorithms.* 1980.

130 P.J.W. ten Hagen, T. Hagen, P. Klint, H. Noot, H.J. Sint, A.H. Veen. *ILP: intermediate language for pictures.* 1980.

131 R.J.R. Back. *Correctness preserving program refinements: proof theory and applications.* 1980.

132 H.M. Mulder. *The interval function of a graph.* 1980.

133 C.A.J. Klaassen. *Statistical performance of location estimators.* 1981.

134 J.C. van Vliet, H. Wupper (eds.). *Proceedings international conference on ALGOL 68.* 1981.

135 J.A.G. Groenendijk, T.M.V. Janssen, M.J.B. Stokhof (eds.). *Formal methods in the study of language, part I.* 1981.

136 J.A.G. Groenendijk, T.M.V. Janssen, M.J.B. Stokhof (eds.). *Formal methods in the study of language, part II.* 1981.

137 J. Telgen. *Redundancy and linear programs.* 1981.

138 H.A. Lauwerier. *Mathematical models of epidemics.* 1981.

139 J. van der Wal. *Stochastic dynamic programming, successive approximations and nearly optimal strategies for Markov decision processes and Markov games.* 1981.

140 J.H. van Geldrop. *A mathematical theory of pure exchange economies without the no-critical-point hypothesis.* 1981.

141 G.E. Welters. *Abel-Jacobi isogenies for certain types of Fano threefolds.* 1981.

142 H.R. Bennett, D.J. Lutzer (eds.). *Topology and order structures, part 1.* 1981.

143 J.M. Schumacher. *Dynamic feedback in finite- and infinite-dimensional linear systems.* 1981.

144 P. Eijgenraam. *The solution of initial value problems using interval arithmetic; formulation and analysis of an algorithm.* 1981.

145 A.J. Brentjes. *Multi-dimensional continued fraction algorithms.* 1981.

146 C.V.M. van der Mee. *Semigroup and factorization methods in transport theory.* 1981.

147 H.H. Tigelaar. *Identification and informative sample size.* 1982.

148 L.C.M. Kallenberg. *Linear programming and finite Markovian control problems.* 1983.

149 C.B. Huijsmans, M.A. Kaashoek, W.A.J. Luxemburg, W.K. Vietsch (eds.). *From A to Z, proceedings of a symposium in honour of A.C. Zaanen.* 1982.

150 M. Veldhorst. *An analysis of sparse matrix storage schemes.* 1982.

151 R.J.M.M. Does. *Higher order asymptotics for simple linear rank statistics.* 1982.

152 G.F. van der Hoeven. *Projections of lawless sequencies.* 1982.

153 J.P.C. Blanc. *Application of the theory of boundary value problems in the analysis of a queueing model with paired services.* 1982.

154 H.W. Lenstra, Jr., R. Tijdeman (eds.). *Computational methods in number theory, part I.* 1982.

155 H.W. Lenstra, Jr., R. Tijdeman (eds.). *Computational methods in number theory, part II.* 1982.

156 P.M.G. Apers. *Query processing and data allocation in distributed database systems.* 1983.

157 H.A.W.M. Kneppers. *The covariant classification of two-dimensional smooth commutative formal groups over an algebraically closed field of positive characteristic.* 1983.

158 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science IV, distributed systems, part 1.* 1983.

159 J.W. de Bakker, J. van Leeuwen (eds.). *Foundations of computer science IV, distributed systems, part 2.* 1983.

160 A. Rezus. *Abstract AUTOMATH.* 1983.

161 G.F. Helminck. *Eisenstein series on the metaplectic group, an algebraic approach.* 1983.

162 J.J. Dik. *Tests for preference.* 1983.

163 H. Schippers. *Multiple grid methods for equations of the second kind with applications in fluid mechanics.* 1983.

164 F.A. van der Duyn Schouten. *Markov decision processes with continuous time parameter.* 1983.

165 P.C.T. van der Hoeven. *On point processes.* 1983.

166 H.B.M. Jonkers. *Abstraction, specification and implementation techniques, with an application to garbage collection.* 1983.

167 W.H.M. Zijm. *Nonnegative matrices in dynamic programming.* 1983.

168 J.H. Evertse. *Upper bounds for the numbers of solutions of diophantine equations.* 1983.

169 H.R. Bennett, D.J. Lutzer (eds.). *Topology and order structures, part 2.* 1983.

CWI TRACTS

1 D.H.J. Epema. *Surfaces with canonical hyperplane sections.* 1984.

2 J.J. Dijkstra. *Fake topological Hilbert spaces and characterizations of dimension in terms of negligibility.* 1984.

3 A.J. van der Schaft. *System theoretic descriptions of physical systems.* 1984.

4 J. Koene. *Minimal cost flow in processing networks, a primal approach.* 1984.

5 B. Hoogenboom. *Intertwining functions on compact Lie groups.* 1984.

6 A.P.W. Böhm. *Dataflow computation.* 1984.

7 A. Blokhuis. *Few-distance sets.* 1984.

8 M.H. van Hoorn. *Algorithms and approximations for queueing systems.* 1984.

9 C.P.J. Koymans. *Models of the lambda calculus.* 1984.

10 C.G. van der Laan, N.M. Temme. *Calculation of special functions: the gamma function, the exponential integrals and error-like functions.* 1984.

11 N.M. van Dijk. *Controlled Markov processes; time-discretization.* 1984.

12 W.H. Hundsdorfer. *The numerical solution of nonlinear stiff initial value problems: an analysis of one step methods.* 1985.

13 D. Grune. *On the design of ALEPH.* 1985.

14 J.G.F. Thiemann. *Analytic spaces and dynamic programming: a measure theoretic approach.* 1985.

15 F.J. van der Linden. *Euclidean rings with two infinite primes.* 1985.

16 R.J.P. Groothuizen. *Mixed elliptic-hyperbolic partial differential operators: a case-study in Fourier integral operators.* 1985.

17 H.M.M. ten Eikelder. *Symmetries for dynamical and Hamiltonian systems.* 1985.

18 A.D.M. Kester. *Some large deviation results in statistics.* 1985.

19 T.M.V. Janssen. *Foundations and applications of Montague grammar, part 1: Philosophy, framework, computer science.* 1986.

20 B.F. Schriever. *Order dependence.* 1986.

21 D.P. van der Vecht. *Inequalities for stopped Brownian motion.* 1986.

22 J.C.S.P. van der Woude. *Topological dynamix.* 1986.

23 A.F. Monna. *Methods, concepts and ideas in mathematics: aspects of an evolution.* 1986.

24 J.C.M. Baeten. *Filters and ultrafilters over definable subsets of admissible ordinals.* 1986.

25 A.W.J. Kolen. *Tree network and planar rectilinear location theory.* 1986.

26 A.H. Veen. *The misconstrued semicolon: Reconciling imperative languages and dataflow machines.* 1986.

27 A.J.M. van Engelen. *Homogeneous zero-dimensional absolute Borel sets.* 1986.

28 T.M.V. Janssen. *Foundations and applications of Montague grammar, part 2: Applications to natural language.* 1986.

29 H.L. Trentelman. *Almost invariant subspaces and high gain feedback.* 1986.

30 A.G. de Kok. *Production-inventory control models: approximations and algorithms.* 1987.

31 E.E.M. van Berkum. *Optimal paired comparison designs for factorial experiments.* 1987.

32 J.H.J. Einmahl. *Multivariate empirical processes.* 1987.

33 O.J. Vrieze. *Stochastic games with finite state and action spaces.* 1987.

34 P.H.M. Kersten. *Infinitesimal symmetries: a computational approach.* 1987.

35 M.L. Eaton. *Lectures on topics in probability inequalities.* 1987.

36 A.H.P. van der Burgh, R.M.M. Mattheij (eds.). *Proceedings of the first international conference on industrial and applied mathematics (ICIAM 87).* 1987.

37 L. Stougie. *Design and analysis of algorithms for stochastic integer programming.* 1987.

38 J.B.G. Frenk. *On Banach algebras, renewal measures and regenerative processes.* 1987.

39 H.J.M. Peters, O.J. Vrieze (eds.). *Surveys in game theory and related topics.* 1987.

40 J.L. Geluk, L. de Haan. *Regular variation, extensions and Tauberian theorems.* 1987.

41 Sape J. Mullender (ed.). *The Amoeba distributed operating system: Selected papers 1984-1987.* 1987.

42 P.R.J. Asveld, A. Nijholt (eds.). *Essays on concepts, formalisms, and tools.* 1987.

43 H.L. Bodlaender. *Distributed computing: structure and complexity.* 1987.

44 A.W. van der Vaart. *Statistical estimation in large parameter spaces.* 1988.

45 S.A. van de Geer. *Regression analysis and empirical processes.* 1988.

46 S.P. Spekreijse. *Multigrid solution of the steady Euler equations.* 1988.

47 J.B. Dijkstra. *Analysis of means in some non-standard situations.* 1988.

48 F.C. Drost. *Asymptotics for generalized chi-square goodness-of-fit tests.* 1988.

49 F.W. Wubs. *Numerical solution of the shallow-water equations.* 1988.

50 F. de Kerf. *Asymptotic analysis of a class of perturbed Korteweg-de Vries initial value problems.* 1988.

51 P.J.M. van Laarhoven. *Theoretical and computational aspects of simulated annealing.* 1988.

52 P.M. van Loon. *Continuous decoupling transformations for linear boundary value problems.* 1988.

53 K.C.P. Machielsen. *Numerical solution of optimal control problems with state constraints by sequential quadratic programming in function space.* 1988.

54 L.C.R.J. Willenborg. *Computational aspects of survey data processing.* 1988.

55 G.J. van der Steen. *A program generator for recognition, parsing and transduction with syntactic patterns.* 1988.

56 J.C. Ebergen. *Translating programs into delay-insensitive circuits.* 1989.

57 S.M. Verduyn Lunel. *Exponential type calculus for linear delay equations.* 1989.

58 M.C.M. de Gunst. *A random model for plant cell population growth.* 1989.

59 D. van Dulst. *Characterizations of Banach spaces not containing* l^1. 1989.

60 H.E. de Swart. *Vacillation and predictability properties of low-order atmospheric spectral models.* 1989.

61 P. de Jong. *Central limit theorems for generalized multilinear forms.* 1989.

62 V.J. de Jong. *A specification system for statistical software.* 1989.

63 B. Hanzon. *Identifiability, recursive identification and spaces of linear dynamical systems, part I.* 1989.

64 B. Hanzon. *Identifiability, recursive identification and spaces of linear dynamical systems, part II.* 1989.

65 B.M.M. de Weger. *Algorithms for diophantine equations.* 1989.

66 A. Jung. *Cartesian closed categories of domains.* 1989.

67 J.W. Polderman. *Adaptive control & identification: Conflict or conflux?.* 1989.

68 H.J. Woerdeman. *Matrix and operator extensions.* 1989.

69 B.G. Hansen. *Monotonicity properties of infinitely divisible distributions.* 1989.

70 J.K. Lenstra, H.C. Tijms, A. Volgenant (eds.). *Twenty-five years of operations research in the Netherlands: Papers dedicated to Gijs de Leve.* 1990.

71 P.J.C. Spreij. *Counting process systems. Identification and stochastic realization.* 1990.

72 J.F. Kaashoek. *Modeling one dimensional pattern formation by anti-diffusion.* 1990.

73 A.M.H. Gerards. *Graphs and polyhedra. Binary spaces and cutting planes.* 1990.

74 B. Koren. *Multigrid and defect correction for the steady Navier-Stokes equations. Application to aerodynamics.* 1991.

75 M.W.P. Savelsbergh. *Computer aided routing.* 1992.

76 O.E. Flippo. *Stability, duality and decomposition in general mathematical programming.* 1991.

77 A.J. van Es. *Aspects of nonparametric density estimation.* 1991.

78 G.A.P. Kindervater. *Exercises in parallel combinatorial computing.* 1992.

79 J.J. Lodder. *Towards a symmetrical theory of generalized functions.* 1991.

80 S.A. Smulders. *Control of freeway traffic flow.* 1993.

81 P.H.M. America, J.J.M.M. Rutten. *A parallel object-oriented language: design and semantic foundations.* 1992.

82 F. Thuijsman. *Optimality and equilibria in stochastic games.* 1992.

83 R.J. Kooman. *Convergence properties of recurrence sequences.* 1992.

84 A.M. Cohen (ed.). *Computational aspects of Lie group representations and related topics. Proceedings of the 1990 Computational Algebra Seminar at CWI, Amsterdam.* 1991.

85 V. de Valk. *One-dependent processes.* 1994.

86 J.A. Baars, J.A.M. de Groot. *On topological and linear equivalence of certain function spaces.* 1992.

87 A.F. Monna. *The way of mathematics and mathematicians.* 1992.

88 E.D. de Goede. *Numerical methods for the three-dimensional shallow water equations.* 1993.

89 M. Zwaan. *Moment problems in Hilbert space with applications to magnetic resonance imaging.* 1993.

90 C. Vuik. *The solution of a one-dimensional Stefan problem.* 1993.

91 E.R. Verheul. *Multimedians in metric and normed spaces.* 1993.

92 J.L.M. Maubach. *Iterative methods for non-linear partial differential equations.* 1993.

93 A.W. Ambergen. *Statistical uncertainties in posterior probabilities.* 1993.

94 P.A. Zegeling. *Moving-grid methods for time-dependent partial differential equations.* 1993.

95 M.J.C. van Pul. *Statistical analysis of software reliability models.* 1993.

96 J.K. Scholma. *A Lie algebraic study of some integrable systems associated with root systems.* 1993.

97 J.L. van den Berg. *Sojourn times in feedback and processor sharing queues.* 1993.

98 A.J. Koning. *Stochastic integrals and goodness-of-fit tests.* 1993.

99 B.P. Sommeijer. *Parallelism in the numerical integration of initial value problems.* 1993.

100 J. Molenaar. *Multigrid methods for semiconductor device simulation.* 1993.

101 H.J.C. Huijberts. *Dynamic feedback in nonlinear synthesis problems.* 1994.

102 J.A.M. van der Weide. *Stochastic processes and point processes of excursions.* 1994.

103 P.W. Hemker, P. Wesseling (eds.). *Contributions to multigrid.* 1994.

104 I.J.B.F. Adan. *A compensation approach for queueing problems.* 1994.

105 O.J. Boxma, G.M. Koole (eds.). *Performance evaluation of parallel and distributed systems - solution methods. Part 1.* 1994.

106 O.J. Boxma, G.M. Koole (eds.). *Performance evaluation of parallel and distributed systems - solution methods. Part 2.* 1994.