INT. J. PROD. RES., 2001, VOL. 39, NO. 2, 205–224

Generalized queueing network analysis of integrated supply chains

N. R. SRINIVASA RAGHAVAN†* and N. VISWANADHAM‡

Supply chain networks are formed from complex interactions between several companies whose aim is to produce and deliver goods to the customers at specified times and places. Computing the total lead time for customer orders entering such a complex network of companies is an important exercise. In this paper we present analytical models for evaluating the average lead times of make-to-order supply chains. In particular, we illustrate the use of generalized queueing networks to compute the mean and variance of the lead time. We present four interesting examples and develop queueing network models for them. The first two examples consider pipeline supply chains and compute the variance of lead time using queueing network approximations available in the literature. This analysis indicates that for the same percentage increase in variance, an increase at the downstream facility has a far more disastrous effect than the same increase at an upstream facility. Through another example, we illustrate the point that coordinated improvements at all the facilities is important and improvements at individual facilities may not always lead to improvements in the supply chain performance. The existing literature on approximate methods of analysis of fork-join queueing systems assumes heavy traffic and requires tedious computations. We present here two tractable approximate analytical methods for lead time computation in a class of fork-join queueing systems. Our method is based on the results presented by Clarke in 1961. For the case where the 'joining' servers of the queueing system are of the type D/N/1, we present an easy to use approximate method and illustrate its use in evaluating decisions regarding logistics (for instance, who should own the logistics fleet—the manufacturer or the vendor?) and computing simple upper bounds for delivery reliability, that is the probability that customer desired due dates are met.

1. Supply chain networks

Manufacturing supply chain networks (SCNs) are formed from complex interconnections between various manufacturing companies and service providers such as raw material vendors, original equipment manufacturers (OEMs), logistics operators, warehouse operators, distributors, retailers and customers (see figure 1). One can succinctly define supply chain management (SCM) as the coordination or integration of the activities of all the companies involved in procuring, producing, delivering and maintaining products and services to customers located in geographically different places. Traditionally, each company performed marketing, distribution, planning, manufacturing and purchasing activities independently,

Revision received July 2000.

† Management Studies, Indian Institute of Science, Bangalore, India 560 012.

‡ Mechanical and Production Engineering, National University of Singapore, Singapore 119260.

* To whom correspondence should be addressed. e-mail: raghavan@mgmt.iisc.ernet.in

International Journal of Production Research ISSN 0020–7543 print/ISSN 1366–588X online
http://www.tandf.co.uk/journals
DOI: 10.1080/00207540010003846

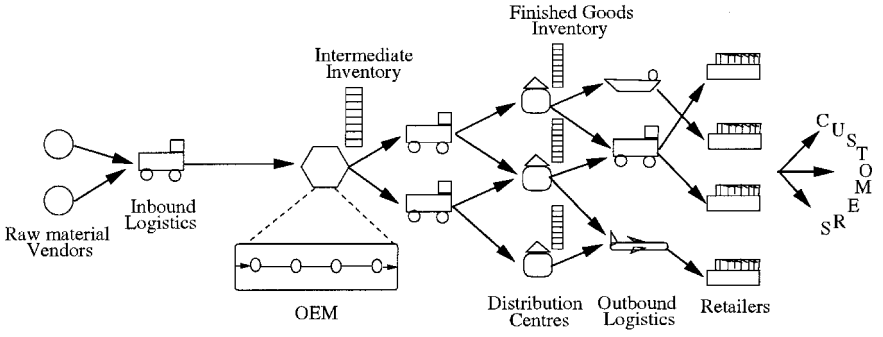

Figure 1. The supply chain network.

optimizing their own functional objectives. SCM is a process-oriented approach to coordinating all organizations and all functions involved in the delivery process.

The product moving through the SCN transits several organizations and, each time a transition is made, logistics is involved. Also, since each of the organizations is under independent control, there are interfaces between organizations and materials and information flows depend on how these interfaces are managed. We define interfaces as the procedures and vehicles for transporting information and materials across functions or organizations such as negotiations, approvals (so called paper work), decision making, and finally inspection of components/assemblies, etc. For example, the interface between a supplier and manufacturer involves procurement decisions such as price, delivery frequencies and nature of information sharing at the strategic level and the actual order processing and delivery at the operational level. The coordination of the SCN plays a large role in the overall functioning of the supply chain process. In most cases, there is an integrator for the network, who could be an original equipment manufacturer, coordinating the flow of orders and materials throughout the network. Modelling and analysis of such a complex system is crucial for performance evaluation and for comparing competing supply chains. In this paper, we view a supply chain as a probabilistic network and present a modelling approach to computing performance measures such as lead time and work in process inventory. In particular, we investigate the use of queueing network models for computing the lead time and other performance measures. In the rest of this section we review the types of SCN and the different order-fulfilment policies.

1.1. *Supply chain network configuration*

The configuration of the supply chain defines the interconnection pattern between its facilities. Suppliers' manufacturing plants, logistics, final assembly plants, packaging centres and transshipment (cross-docking) points are examples of facilities. Not all SCNs have the same configuration. Depending on the product structure, several network configurations are possible. One could identify four major supply chain configurations, namely serial, converging, diverging and converging–diverging networks. The supply chains can be operated in different ways: make-to-stock, make-to-order or assemble-to-order. We briefly cover these issues below.

(1) *Serial structure*. Here, one facility of the network feeds into another and the entire supply chain resembles a single pipeline.

(2) *Divergent structure*. This structure resembles a cone. At the vertex of the cone is the facility that produces a base product from which are produced the derivatives. The petroleum industry is a typical case in point. Distribution intensive industry supply chains often have a divergent structure.
(3) *Convergent structure*. In this case, there is a series of sub-assembly stages finally leading to a finished product as in the case of automobiles and air crafts.
(4) *Network structure*. This is a tandem combination of convergent and divergent structures as in the computer industry which is both sourcing and distribution intensive.

In this paper, we consider the serial and convergent structures, but the application of the methodology to other structures is straightforward.

1.2. *Operational models*

Another important aspect of the supply chain operation is the supply chain planning and control (SPC) methodology. A customer order for a product triggers a series of activities in the supply chain facilities, and these have to be synchronized so that the end customer order is satisfied. The SPC specifies the business model and hence determines the paths for the information and material flow in the supply chain. There are three broad models followed in practice.

(1) *Make-to-stock* (MTS). Here, the end customer products are satisfied from stocks of an inventory of finished goods that are kept at various retail points of the SCN.
(2) *Make-to-order* (MTO). In this SPC technique, a *confirmed* customer order triggers the flow of materials and information in the supply chain. Each customer order is unique in terms of manufacturing, procuring, packaging or logistics requirements. There is very little or no inventory maintained of the finished goods or component materials.
(3) *Assemble-to-order* (ATO). Here, a variety of products are assembled to order from components and sub-assemblies, which are either manufactured-to-stock or outsourced. One crucial issue in such a business model is the location of the customer order decoupling point, a point in the supply chain until which sub-assemblies are made to forecast and beyond which products are built to order.

The SPC technique defines the inventory control rules followed by the supply chain. Each facility of the supply chain could follow its own inventory control policies and ways of handling the sourcing and delivery functions. It is possible that not all the facilities produce to stock. Some facilities might assemble goods to order. The crucial issues of when to order and how much to order define these policies. For instance in base stock policies, one unit (alternatively, a stock keeping unit (SKU)) of inventory is replenished as soon as a unit of goods held at the facility is depleted. On the other hand, if the facility is following a reorder point based policy, it replenishes items as soon as a preset reorder level is reached, ordering each time such that a targeted level of inventory is reached. For a detailed discussion, refer to Vollman *et al.* (1998).

1.3. *Literature survey*

In this section, we briefly survey the literature on mathematical models for supply chains. The analytical modelling of SCNs can be classified broadly into two areas: network design and performance analysis methods. The network design models help in strategic and tactical decision making. They are basically mixed-integer programming models and are used to decide what products to produce and for what markets, where and how to produce them, and using what resources. There is a large amount of literature on this subject and a number of survey papers have also appeared on this subject (see Erenguc *et al.* (1999), and references cited therein). There is also a large body of literature in the development of multi-echelon inventory control models. A comprehensive review of these models can be found in the book by Silver *et al.* (1998). Other studies include location of production and inventory facilities (Arntzen *et al.* 1995, Sankaran and Srinivasa Raghavan 1997) and choosing make-to-order or make-to-stock policies (Connors *et al.* 1995). Performance analysis of SCNs is basically conducted to determine the lead time, variation, cost, reliability and flexibility. SCNs are discrete event dynamical systems (DEDSs) in which the evolution of the system depends on the complex interaction of the timing of various discrete events such as the arrival of components at the supplier, the departure of the truck from the supplier, the start of an assembly at the manufacturer, the arrival of the finished goods at the customer, payment approval by the seller, etc. The state of the system changes only at discrete events in time. Over the past two decades, there has been a tremendous amount of research interest in this area. Very attractive higher-level general purpose simulation packages are now available that can faithfully model the value delivery processes of a manufacturing enterprise. These include SIMPROCESS, PROMODEL and Taylor II. The simulation of a SCN involves developing a simulation model, coding it, validating it, designing the experiments, and finally conducting a statistical analysis to obtain the performance measures.

1.4. *Analytical models*

Our aim here is to summarize five analytical techniques useful for modelling SCNs, namely series parallel graphs, Markov chains, Petri nets, queueing networks and system dynamics models.

1.4.1. *Series parallel graphs*

Series parallel graphs can model a SCN by assigning probability distributions to the lead time of the activities in the graphs. These graphs, show the precedence and concurrency of the activities of the material and information flow. Assuming that all the activities are statistically independent, one can determine the mean and variance of the lead times. See Viswanadham and Srinivasa Raghavan (1999) for details for this approach in the context of SCNs.

1.4.2. *Markov chains*

The use of Markov models in the study of performance of manufacturing systems (Viswanadham and Narahari 1992) is well known. Direct modelling of a SCN as a Markov chain would be very difficult and expensive. Higher-level models based on Petri nets and queueing networks are ultimately solved as Markov chains using software packages such as SPNP.

1.4.3. *Petri nets*

Faithful modelling of iteration, synchronization, forks and joins that arise in SCNs is possible using Petri nets. If too detailed models are developed, however, numerical solution may turn out to be a nightmare. The hierarchical modelling discussed in this paper provides a tractable way of handling largeness here. In Srinivasa Raghavan and Viswanadham (1999), we used Petri nets to model supply chains and obtained some interesting results. As pointed out in that reference, the exponential processing time assumption limits its applicability to real world systems.

1.4.4. *Queueing networks*

The most general SCN can be modelled as a fork-join queueing network model with iteration or re-entrancy. An analytical solution of these general models is not available, and approximations are available in special cases only. Some solutions can be found in Srinivasa Raghavan (1998). This is an area of active research.

1.4.5. *System dynamics models*

Here the SCN is modelled using differential equations. Forrester (1961) first explained the 'bull whip' effect using these models. There is much literature using this approach for real world system modelling.

1.5. *Models for lead time computation*

The lead time of an order entering the supply chain is the total time it spends in the supply chain and is a crucial performance measure. All the models discussed above can be used to arrive at the total expected supply chain lead time and its variance. An estimate of the mean and variance of the supply chain lead time can help one in quoting the delivery time reliably for a given customer order. In this paper, we treat the lead time computation problem for a class of make-to-order supply chains as multi-class open generalized queueing networks. We utilize the existing efficient approximation algorithms to compute the expected supply chain lead time and variance. This is the subject of § 2. We remark that existing literature on approximate methods of analysis of fork-join queueing systems assumes heavy traffic (Nguyen 1993, 1994) and requires tedious computations. For a class of fork-join queueing systems, we present in § 3, a new approximate method of analysis which we use in the analysis of supply chains. For the case of the joining servers of the queueing systems of the type D/N/1, we present an easy to use approximate method, based on the results of Clarke (1961). We report encouraging results for this class of fork-join system, with some possible applications in the supply chain context. We conclude the paper in § 4.

2. Generalized queueing network models of some SCNs

In this section, we present a method for evaluating the performance of make-to-order supply chains using general queueing networks (GQNs). In the dynamic case the orders for end products arrive in a random manner and demand a random processing time at each of the facilities. We consider and model a real world example of a manufacturer of lubrication systems for OEMs of earth movers. In this example a *single* manufacturer supplies several different types of end products to the OEM. It may be noted that the above configuration follows the make-to-order kind of production planning and control.

2.1. *A note on approximate analysis*

Solving a network of queues for determining the steady state average waiting times and mean queue lengths is a well researched area. If the arrivals are Poisson and service times are exponentially distributed, then we use either Jackson's results or Gordon-Newell's results, depending on whether the queueing networks are open or closed, respectively. See for instance Buzacott and Shantikumar (1993) for an elaborate treatment on this issue. On the other hand, if the arrivals and service times are non-exponential, and are generally distributed with given mean and variances (or squared coefficients of variation), then seeking exact analytical expressions for computing the performance measures of interest is a futile exercise unless the distributions have certain 'nice' features. In general, one resorts to approximate methods of analysing the queueing networks, where each node is G/G/m type. Whitt (1983) proposed a queueing network analyser (QNA), a decomposition based method for analysing the generalized queueing networks. The standard QNA is based on the decomposition of the considered queueing network into GI/G/m queues. Therefore, arrival and service processes are regarded as renewal processes characterized by the mean and the squared coefficient of variation. It is possible to determine the internal flow between the nodes of the network and to decompose the network into GI/G/m queues. These queues can be analysed independently of each other and for each queue the performance measures can be calculated. After composing the network again it is possible to determine the performance measures for the entire network. In contrast to other well known analytical methods such as Jackson networks, the QNA can handle general distributed inter-arrival and service times. It is not restricted to exponential distributions. Therefore the queueing model represents real networks much better than exact analytical methods can do. Besides, the QNA offers more results than exact analytical methods do. Apart from the mean values, it also determines variance of the performance measures, which is important in real world supply chains. Furthermore, the calculation is quite fast so that even big networks can be analysed in an appropriate time. For the experiments reported in this paper, we ran the GQN models using a package developed locally (Kumar 1994) for analysing the queueing models. This software makes use of results due to Whitt (1983).

2.2. *A two-tier manufacturer producing multiple end products*

Consider the supply chain network shown in figure 2. The manufacturer M produces three different types of lubrication system. All the product types go through five operations. Operations O_1, O_2 and O_3 are performed on the factory floor of the manufacturer M. The remaining two operations are outsourced to second-tier suppliers ($S_1 \ldots S_6$). The operations between O_1 and O_2 are outsourced to suppliers S_1, S_2 and S_3 depending on the type of lubrication system on production. Similarly, the operations between O_2 and O_3 are outsourced to suppliers S_4, S_5 and S_6. Thus, every alternate operation of all end products goes out of the factory floor of M. This involves outbound logistics each time, arranged by M to fetch semi-finished product from the sub-contractors. We assume that M is in a make-to-order environment wherein each arriving order for a class of product is treated as a batch of products of that type.

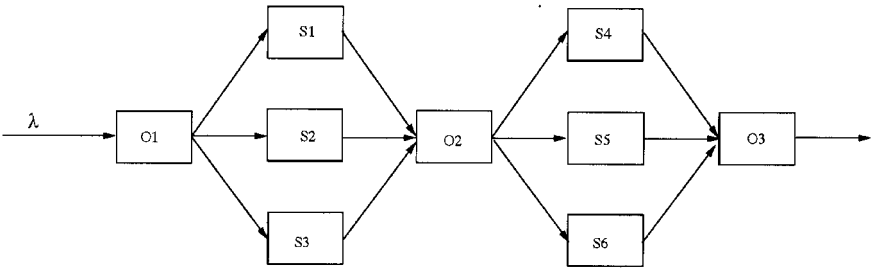

Figure 2. The supply chain for the first example.

2.2.1. *A GQN model for the above supply chain*

In order to conduct performance analysis of the above supply chain, we model the various enterprises in the above supply chain (for instance, $S_1 \ldots S_6$) by 'servers' of an open queueing network. We also model the operations in the factory M (O_1 etc.) as servers of the above queueing network. Jobs arriving at these servers are enqueued at these facilities, queueing space being unbounded. One can conduct a survey on the fraction of jobs going from operation O_1 to suppliers S_1 through S_3. These will give us the transition probabilities for the queueing network. The service processes at the various servers are assumed to follow general distributions whose mean and variance are known *a priori*. Thus, we are ready to conduct a steady state analysis of the above network. We use hypothetical data for arrival rates, service rates and transition probabilities, since our intention is to bring out the modelling power only of queueing networks. To simplify the analysis, we aggregate the three products into one single type and take the effective arrival rate as the sum of the three components, with the squared coefficient of variation (SCV) of arrivals computed from the components. (Alternatively we can study the multi-class queueing network using approximate methods for performance analysis.) The input parameters for the base case are shown in table 1.

We study the effect of

(1) increasing the SCV of the inter-arrival rate;

	Service rate (units per hour)	
Server name	Mean	SCV
O_1	120	0.5
S_1	40	0.5
S_2	50	0.5
S_3	60	0.333
O_2	100	0.5
S_4	30	0.5
S_5	40	0.5
S_6	50	0.5
O_3	180	0.5

$\lambda_1 = 20$, $\lambda_2 = 30$, $\lambda_3 = 40$ units per hour; SCV = 0.5 each. Hence, $\lambda = 90$, SCV = 0.179.

Table 1. The base case input parameters for the supply chain of figure 2.

(2) increasing the service rate (mean and SCV) of one of the outsourced suppliers; and
(3) increasing the capacity of the bottleneck server, which happens to be O_2 (the operation performing the second in-plant operation).

The above three effects are reflected in tables 2–5. For brevity, we consider only the waiting time as the major performance measure. (Trends for average queue lengths, or work-in-process (WIP), are similar.)

2.2.2. *Remarks*

As is clear from tables 2–5, we observe the following.

(1) The SCV of the input arrival rate has a definite influence on the cycle time. It is essential to control the input arrival rate to reduce the cycle times. This needs the manufacturer to engage in closer ties with customers, who generate the orders (see table 2).
(2) A counterintuitive result is seen in table 3. When we reduce the service rate of the (outsourced) supplier S_3, the cycle times for product 3 increase. This is not surprising, but as a side effect, the cycle times of the other two products have actually reduced! An explanation for this is that the succeeding in-plant operation O_2 and all the downstream servers, obtain products of type 1 and type 2 more frequently, since product 3 is slowed down. The manufacturer has to negotiate appropriately the supply lead times with the sub-contractors, keeping in view interactive effects such as the above. On the other hand, the effect of the SCV of the service rate of S_3 is obtained as expected, namely, with an increase in SCV (keeping the mean constant at 60), all the average cycle times increase (see table 4).
(3) The bottleneck server being O_2, we found that an increase in its service rate naturally results in better performance. Comparing columns 2 and 3 of table 5, we observe that it is enough if the service rate is increased by some value (here, 105) beyond which the returns are at diminishing rates.

	SCV of aggregate arrival rate		
Product type	0.179	0.253	0.403
1	0.1387	0.1391	0.1399
2	0.1409	0.1415	0.1428
3	0.1424	0.1432	0.1449

Table 2. Effect of the SCV of arrival rates on average cycle times (hours).

	Service rate of S_3			
Product type	70	60	50	45
1	0.1389	0.1387	0.1383	0.1380
2	0.1413	0.1409	0.1403	0.1398
3	0.1331	0.1424	0.1695	0.2230

Table 3. Effect of service rates of S_3 on average cycle times (hours).

	SCV of service rate of S_3		
Product type	0.333	0.5	1.0
1	0.1387	0.1391	0.1403
2	0.1409	0.1415	0.1433
3	0.1424	0.1456	0.1555

Table 4. Effect of SCV of service rates of S_3 on average cycle times (hours).

	Service rate (mean, SCV) of O_2		
Product type	(100, 0.5)	(105, 0.5)	(120, 1.0)
1	0.1387	0.1345	0.1339
2	0.1409	0.1347	0.1342
3	0.1424	0.1341	0.1338

Table 5. Effect of service rates of the bottleneck, O_2, on average cycle times (hous).

(4) We have also analysed the effect of slowing down of the downstream server, even though the upstream server is faster. For example for product 3, we increased the service rate of S_3 from 60 (base case) to 70, simultaneously decreasing that of S_6 from 50 to 45. We see that the average cycle times for the three product types are, respectively, 0.1390, 0.1414 and 0.1993. Comparing this with the values in table 3, first column (where the downstream server is as in the base case), we note that the average cycle times of *all* the products have gone up. This re-emphasizes our claim that, *any attempts at minimizing individual supplier performance in isolation are bound to yield sub-optimal results*. Thus the need for an integrated approach to tuning the cycle times of the members of the supply chain.

2.3. *Influence of variability*

In this section, we present an example to illustrate the effect of variability reduction on the performance of a make-to-order supply chain. Specifically, we consider its impact on the average lead time and the average WIP inventory. We note that variability in processing times and arrival patterns causes congestion at various facilities of the supply chain. Also, it is known that it is always better to reduce the variability at its source. For example, if we have a tandem network of facilities, the best results are achieved if the variability in processing times of the first (set of) facility(s) is reduced.

In the context of supply chains, this means that reducing the variability of the lead times of the factory is not enough if, for instance, the supplier keeps supplying the needed raw materials with high lead time variability and the distribution system has highly variable lead times for supplying the finished goods to the end customers. The *process improvement* program of the factory concerned may well yield poor results, from the end customer's viewpoint. This is indeed verifiable by using GQN models for the supply chain. Consider the GQN model of a supply chain shown in figure 3. We now investigate the effect of decreasing the variability of the processing times at the suppliers, the factory floor and the outbound logistics.

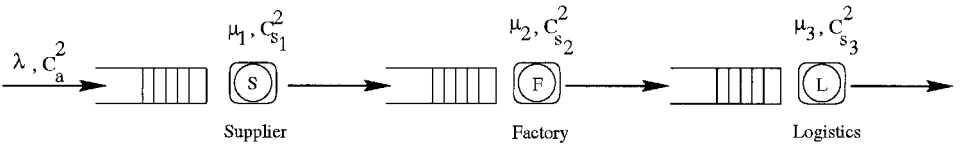

Figure 3. GQN model for illustrating the effect of variability reduction.

Facility	Service rate	SCV
Supplier	30.0	0.8
Factory	20.0	0.8
Logistics	25.0	0.8

Table 6. Input parameters for the supply chain of figure 3. External arrival rate: 15 per hour, SCV: 0.5.

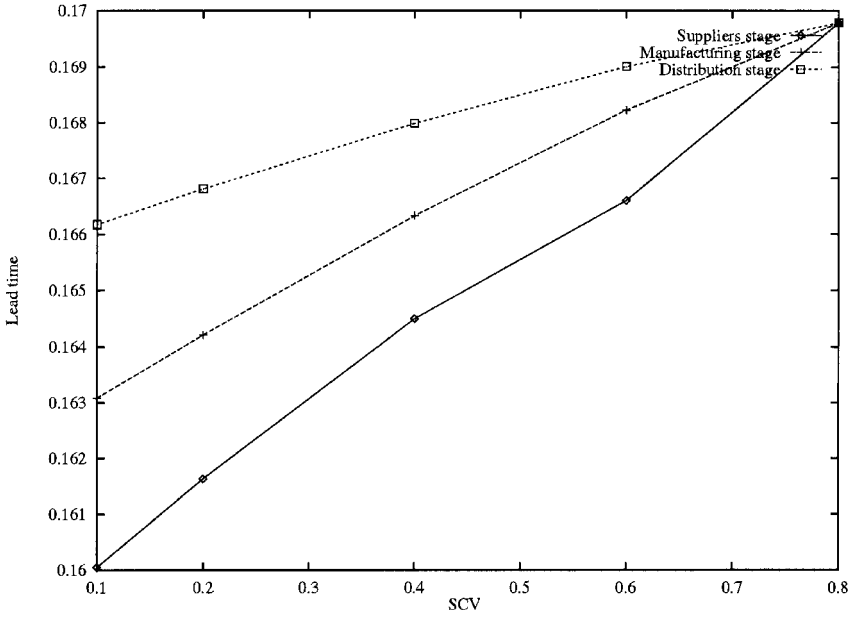

Figure 4. Effect of variability reduction at various stages on lead time.

The variability of the times is captured by the SCV. The parameters for the base case are shown in table 6 and the results are depicted in figure 4. We see that the effect of reducing the SCV of the suppliers while keeping the other two SCVs at the base level, is marked when compared with the cases where the variabilities at the factory or the outbound logistics are altered. This is seen in the shape of the graphs for lead time. (We found that the trends were similar for WIP.) Although we have run the model for a simple pipeline supply chain, it is easily extendable to supply chains of various other configurations, including fork-join networks.

We now present a new method of approximate analysis of fork-join queueing systems in § 3.

3. Approximate analysis of fork-join queueing networks

In this section, we present an approximate method for the performance analysis of certain make-to-order supply chains with a fork-join queueing network (FJQN) structure. Consider the following supply chain (for a detailed treatment, refer to § 3.4 and figure 6).

- There is one end product or a product group that is made to order. Thus we can handle a single product or all products belonging to a particular family.
- Let this product (family) be manufactured from two major sub-assemblies supplied by two different suppliers. The inbound logistics is managed by the suppliers themselves.
- The sub-assemblies are then joined at a manufacturing plant. There is a synchronization wait at this manufacturing plant, for both the sub-assemblies to arrive.
- Since the end item is made-to-order, there is forking at the supplier end, that is, orders for the sub-assemblies are placed simultaneously with the suppliers. This is a structural feature that simplifies the analysis of the underlying FJQN, which will otherwise become unstable. See Gershwin (1994) for an explanation.
- The assembled end product (family) is then delivered to distributors.

We are interested in computing the end-to-end delay, or the total supply chain lead time by its first two moments. It is well known (Kim and Agarwala 1989) that FJQNs are difficult to analyse exactly. Hence many approximation algorithms have been proposed in the literature (see the references in Mascolo *et al.* (1996), Kumar and Shorey (1993), and Baccelli *et al.* (1989)). Exact results are available for the case where the arrivals are Poisson only, with exponential service times and just two joining nodes. See Flatto and Hahn (1984) and Nelson and Tantawi (1988) for details. Most of the approximate methods assume the following.

- The processing times at all nodes of the network belong to the exponential family of distributions (exponential, Erlangian or hyper-exponential).
- The buffer sizes at the various queues are all bounded.
- Studying the *mean* cycle time alone is sufficient.

Since in the case of supply chains, it is common to encounter more general distributions, it is necessary to include general service times in the analysis. Also, the buffer sizes need not necessarily be bounded. Though mean cycle times capture the steady state behaviour, it is necessary to compute the variance of the cycle times too. Before we go into the complete analysis of the supply chain, we describe our approach for the fork-join structure with normally distributed service and deterministic inter-arrival times. In the approximation that we will be developing in this section, we assume that the joining nodes have Gaussian service time distributions with their means at least three times the standard deviation and, in real life supply chains, this assumption is found to be adequate and reasonable (Lee and Billington 1995).

3.1. *Computing the SCV of inter-departure time*

Consider two servers with general service times and infinite buffer sizes as shown in figure 5. Let the arrival process to these be generally distributed, with a fork on every arrival. Let there be a join immediately after the two servers complete service. We are interested in computing the approximate departure process *after* the join, by its first two moments. Observe that the departure process from the join station is the arrival process to any downstream server. Once the moments of the above departure process are computed, the GQN analysis of the remaining part of the queueing network is at hand. In table 7 we detail the notation used. The various times mentioned in the notation are all random variables. Their averages will be denoted by $\mathbb{E}[.]$. It is worth noting that when the fork-join structure is under steady state, the departure rate out of the join node will be equal to the arrival rate at the fork node. Hence it is enough if we get an approximation for the variance of the departure process from the join node. We ignore the effect of blocking of servers and proceed with the (first) assumption that the mean inter-departure rate is the same as the mean inter-arrival rate for each server in isolation. Thus, when there is a fork to the two servers, D_1 and D_2 will have the same means. In order to compute the second moment of $D_1 \ldots D_K$, we analyse servers $S_1 \ldots S_K$ as independent GI/G/1 queues,

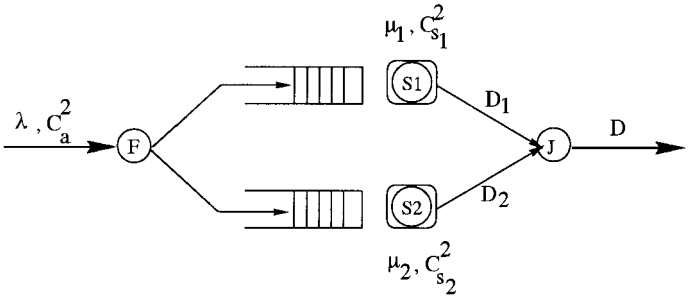

Figure 5. The fork-join structure.

λ and C_a^2	The mean rate and SCV of the arrival process
μ_i	The mean service rate at server S_i, $i = 1, K$
$C_{s_i}^2$	The SCV of service time at server S_i, $i = 1, K$
ρ_i	Utilization of node i, $i = 1, K$
T_i	The sojourn time at server S_i, $i = 1, K$
T_i^*	The maximal value (order statistic) of T_i, $i = 1, K$
D_i	The inter-departure time from server S_i, $i = 1, K$
$\sigma_{D_i}^2$	Variance of D_i, $i = 1, K$
$C_{D_i}^2$	SCV of D_i, $i = 1, K$
D	The inter-departure process after the join node
T	The sojourn time in the fork-join structure
n	Subscript representing the nth job in the system
μ and σ^2	Mean and variance of D

Table 7. Notation for the approximation method.

with the *same* arrival rates as the given external arrival rate, prior to forking. This analysis would give us the mean and SCV of inter departure times from each server. Towards this end, we use the first approximation for the mean cycle time and SCV, given in Buzacott and Shantikumar (1993, pp. 75–76), shown below:

$$C_{D_i}^2 = (1-\rho_i^2)\frac{C_a^2 - \rho_i^2 C_{s_i}^2}{1+\rho_i^2 C_{s_i}^2} + \rho_i^2 C_{s_i}^2, \quad i = 1, \ldots, K, \tag{1}$$

$$\rho_i = \frac{\lambda}{\mu_i}, \quad i = 1, \ldots, K. \tag{2}$$

By our first assumption, $\mathbb{E}[D_1] = \cdots = \mathbb{E}[D_K] = 1/\lambda$. On an average, it is the server (say, $\tilde{k}$) with the greatest mean flow time ($= \max_i T_i$) which is expected to delay the other jobs. Thus the server with the greatest average processing time (and hence its variance) contributes most to the variance of the inter-departure process after the join node. Hence, we compute $C_{D_i}^2$, $i = \tilde{k}$ and choose this value as an approximation for the SCV of the inter-departure process *after* the join node, that is, $C_D^2 = C_{D_{\tilde{k}}}^2$. Thus the departure process *after* the join, namely, D, is computed by its first two moments. We are now ready for the next stage of our aggregated approximate analysis of the entire supply chain. The case of unsymmetric fork-joins also can be studied, that is, if a fork leads to splitting into, say, a tandem network for the one arm and a single server for the other, we can still analyse the same using the above method.

3.2. *Mean waiting time for the fork-join structure*

We observe here that $T_1 \ldots T_K$ can be computed by their first two moments using standard GI/G/1 analysis. Specifically (Buzacott and Shantikumar 1993), we used the following approximations:

$$\mathbb{E}[T_i] = \left\{\frac{\rho_i^2(1+C_{S_i}^2)}{1+\rho_i^2 C_{S_i}^2}\right\}\left\{\frac{(C_a^2+\rho_i^2 C_{S_i}^2)}{2\lambda(1-\rho)}\right\} + \frac{1}{\mu_i}, \quad i = 1, \ldots, K. \tag{3}$$

Thus the mean waiting time for the fork-join construct is given by $\max(T_1, \ldots, T_K)$. There are results for upper bounds for this performance metric; see for instance Kumar and Shorey (1993) and the references therein. Essentially, these bounds are achieved by applying some stochastic ordering properties of associated random variables. Specifically, computable upper bounds are obtained only in the case of such systems wherein the *distribution* of the server sojourn times is known. For the case where this is not known, it is, in general, difficult to obtain bounds. Recently, some interpolation and diffusion approximations have been tried out on symmetric fork-join systems with generally distributed arrival and service times (Nguyen 1993, 1994, Varma and Makowski 1994). We note that such methods again involve tedious numerical computations and are valid under heavy traffic conditions only. For the case where the joining servers have service times that are normally distributed and when the arrival pattern is deterministic, we present below an approximation scheme for computing the mean waiting time. As discussed earlier, once the first two moments of the waiting times at the joining nodes are computed, we assume that the waiting times are independent of each other, and that they are normally distributed with the mean and standard deviation as computed. Although this is a distributional assumption on the waiting times, since we are interested in an approximate

method of analysis, we assume as such. The first four moments of the maximum of n independent normally distributed random variables can be obtained using the approximation detailed in Clarke (1961). We reproduce Clarke's result for the two random variables case:

$$\mathbb{E}[D] = \mathbb{E}[D_1]\Phi(\alpha) + \mathbb{E}[D_2]\Phi(-\alpha) + a\phi(\alpha), \tag{4}$$

$$\mathbb{E}^2[D] = (\mathbb{E}[D_1]^2 + \sigma_{D_1}^2)\Phi(\alpha) + (\mathbb{E}[D_2]^2 + \sigma_{D_2}^2)\Phi(-\alpha) + (\mathbb{E}[D_1] + \mathbb{E}[D_2])a\phi(\alpha), \tag{5}$$

$$\alpha = \frac{1}{a}(\mathbb{E}[D_1] - \mathbb{E}[D_2]), \tag{6}$$

$$a^2 = \sigma_{D_1}^2 + \sigma_{D_2}^2 - 2\sigma_{D_1}\sigma_{D_2}\varrho. \tag{7}$$

In the above equations, ϕ is the standard normal density function and Φ is the corresponding cumulative distribution function. Also, ϱ is the coefficient of correlation between the variables D_1 and D_2, which is assumed to be 0 in our case, owing to the independence assumption. The above formulae can be easily extended for the n independent normal random variables case. Observe that $\max(a, b, c) = \max(a, \max(b, c))$. We use the above formulae and obtain the mean flow time at the fork-join stage.

3.3. *Numerical results*

For purposes of validation, we present the results of using our approximation on two single stage systems, one containing two servers and the other containing ten servers. In order to test our approximation for the SCV of the departure process after the join node and the mean flow time at the fork join structure, the input cases are illustrated in table 8. In the table, the mean inter-arrival time is specified for varying utilization values of the server with the maximum service time of 380 s. The utilization value considered were 50, 80 and 90% respectively.

In table 9, Case B refers to the case where the servers are homogeneous with service times equal to 380 s, for both two and ten server systems. Case C refers to heterogeneous servers with same mean service times as Case A, but their standard deviations decrease from 50 s (for the server with mean service time 200 s) to 5 s (for the server with mean service time 380 s) in steps of 5 s. Case D is similar to Case C, but the service time standard deviations now decrease from 150 s to 60 s in steps of 10 s as in Case C. Note that in Case D, we allow for, in theory, processing times to go negative since mean service time is sometimes less than three times the standard deviation. This is just to verify our approximation. The results are tabulated in table 9. The maximum absolute error percentage from the table shown is found to be 12. We note that this occurs for Case B (with ten homogeneous servers) when the utilization value is 90% . Also, Clarke (1961) showed that the approximation for the

P_{S_i}, $i = 1, \ldots, K$, mean processing times (Gaussian) (s)	200 to 380 in steps of 20
Standard deviation of all processing times	$\frac{1}{4}$ of mean
SCV of all processing times	0.0625
Mean inter-arrival time (s)	760, 475 and 422.2

Table 8. Input parameters (Case A) for the supply chain of figure 6.

		Mean flow time at fork-join stage (s)			
		Case A		Case B	
No. of servers	Utilization (%)	Exact	Approximation	Exact	Approximation
2	50	382.96	382.07	433.04	433.58
	80	394.30	385.26	449.70	437.15
	90	444.88	414.78	514.31	476.16
10	50	450.15	460.00	526.05	524.77
	80	461.50	460.80	550.73	528.80
	90	498.21	477.68	670.85	587.52
		Case C		Case D	
10	50	380.00	381.6	459.36	447.48
	80	380.14	381.6	461.62	448.00
	90	380.30	381.6	474.49	450.53

Table 9. Validation results for single stage fork-join queueing systems.

maximum value of n normally distributed random variables is error prone, especially when the random variables considered have the same mean and variance. This is precisely the case when we consider homogeneous servers in the fork-join stage. On an average, the approximation was found to give absolute error percentages of less than 4%. The absolute error percentage was computed as the difference between the approximated flow time and the time computed from simulation, divided by the time computed from simulation. Based on the results, we can safely conclude that for the best results, the approximation is to be used in moderate utilization conditions (of less than about 85%), while at higher utilizations, it is bound to give absolute errors greater than 10%.

3.4. *Approximate analysis of supply chains*

As we have already discussed, it is common to find (demand) order batching in SCNs. By this we mean that demand from one echelon member to its immediate predecessor arrives at predetermined epochs (say, weekly, monthly, daily, fortnightly, etc.). But each arriving demand may need a random amount of processing at the facilities. Thus, if we consider a supply chain where all the members work on a make-to-order basis, we can view the supply chain as a general open fork-join queueing network with demand arrivals as deterministic, while processing times at the facilities are normally distributed. (We assume that the processing times are such that the probability of getting a negative value for the same is less than 0.05. That is, the mean processing time is at least three times the standard deviation.) Further, if we assume that there is only one stage where the fork-join occurs, then we are in a position to use the approximation that we developed in the earlier section as follows. Consider the supply chain shown in figure 6. Once the arrival process (D) to the first node *after* the join is computed by its first two moments, we use standard methods to further the analysis. We used a software package called RAQS (Kamath 1997) for this. As shown in figure 6, we consider two suppliers S_1 and S_2 with their own logistics L_1 and L_2 respectively, supplying two sub-assemblies to be assembled at the final assembly plant (FAP), at M. We consider the make-to-order material

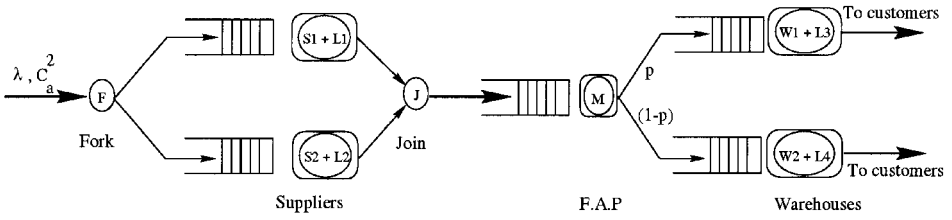

Figure 6. The supply chain network (FJQN) considered for study.

movement philosophy throughout the supply chain. Since every arriving order requires both the sub-assemblies S_1 and S_2, we will have a fork to both the suppliers, and a join immediately following the logistics operations. We assume that each arriving order is for a batch of the end product and each batch is transported by a single logistics carrier. After the final assembly at M, the order splits probabilistically to the retail outlets W_1 and W_2, with probabilities p and $(1-p)$ respectively. These probabilities can indeed be fixed as the percentage of supplies that are rationed to the retailers from the manufacturer. We assume that the service rates of the various facilities including the logistics are specified *a priori*. These values can be obtained after the analysis of the dynamics of each facility in isolation, using generalized Jackson networks. We are not concerned with this here.

We use the approximation that we have developed in the previous section, to get the estimates of the moments of the arrival process at the final assembly plant. This is then used to do a GQN analysis, as in previous sections, of the remaining part of the supply chain network. The major performance measures of interest are:

- the steady state net work-in-process (WIP) inventory of the portion of the supply chain starting from the suppliers; and
- the steady state average waiting time for a job entering the supply chain.

Note that the net WIP for the supply chain is the sum of the inventories of all the joining nodes and the inventories at the nodes downstream of the joining node. These values are obtained, as said before, using the approximations given in Buzacott and Shantikumar (1993) and using RAQS (Kamath 1997). We compare the results due to our approximations and those obtained by a simulation of the supply chain. We used Taylor-II for simulation. (The simulation results detailed below were obtained after conducting several runs for each case and averaging them over all the runs. Also, we resorted to initial transient deletions so that our results pertain to the steady state physics alone. In general, the 95% confidence intervals for the performance measures were found to be within 5%.) For varying values of the arrival rates (ensuring that the FJQN is stable), we computed the approximate departure process from the join node.

Table 10 shows the input parameters for the supply chain shown in figure 6. In this table, a *day* is equal to 24 hours. We assume that all facilities work round the clock. (This assumption does not in any way restrict the application of our approximation.) Also, we considered the SCVs of the service times to be 0.111, which essentially implies that the mean is at least three times the standard deviation. This enables us to use normal random variates in the simulations. We obtain the results for varying utilization levels, which are obtained by varying the arrival rates.

λ and C_a^2	1/day deterministic
$\mu_{S_1+L_1}$ and $\mu_{S_2+L_2}$	2/day, 1.5/day
$C^2_{s_1+l_1}$ and $C^2_{s_2+l_2}$	0.11, 0.11
μ_M and C_M^2	3/day, 0.11
$\mu_{W_1+L_3}$ and $\mu_{W_2+L_3}$	2/day, 2.5/day
$C^2_{w_1+l_3}$ and $C^2_{w_2+l_4}$	0.11, 0.11
Probability of splitting from M to warehouses	0.5 each

Table 10. Input parameters (base case) for the supply chain of figure 6.

For validating the results for a network, say N, we proceed as follows. We assume that there is only a single stage FJQN system which is succeeded by a GQN without any further fork-join. Let us denote the FJQN part of the entire network as N_1 and the rest of the network as N_2. We use the approximation developed in the earlier sections to compute the sojourn time in N_1, and the arrival process to N_2. Thus for purposes of comparison, we simulate only N_1 and use existing algorithms for N_2 as in RAQS. We then aggregate the result for N, observing that sojourn times as also WIP at N_1 and N_2 are additive. Refer to table 11 for such a comparison. We find that the errors while calculating the performance measures starting from the suppliers downstream of the supply chain, are less than 3%. Although we have shown the validity of our results for a simple example, we can use the approximation, as discussed earlier in this section, for the case where the number of suppliers is ten or more.

3.5. *Applications of the approximation*

Continuing on similar lines, we present certain critical decisions in the supply chain which one can make using our method.

3.5.1. *Logistics decisions*

We should like to evaluate the influence of logistics owned by the manufacturer and logistics owned by the suppliers themselves, in terms of the two performance measures of interest, namely, the average inventory and the average lead time. The manufacturer owned logistics reflects in the FJQN directly as follows. The joining nodes *do not* include the logistics service times. This will be included in a downstream server meant for logistics. We evaluated the influence of the same, under different utilizations of the joining nodes. Note that table 11 gives the supplier owned logistics case. For purposes of comparison, the logistics server of figure 7 has an aggregate mean service time equal to the sum of the two logistics times of figure 6, and is normally distributed, as are the two joining nodes. Also, the SCVs of the logistics

λ	Mean supply chain lead time (h)			Average WI		
(order/h)	approximation	exact	error (%)	approximation	exact	error (%)
1.0	2.040	2.049	−0.444	2.002	2.001	+0.050
1.2	2.066	2.090	−1.163	2.271	2.270	+0.044
1.4	2.438	2.379	+2.463	2.947	2.946	+0.034

Table 11. Comparison of approximation with simulation for the supply chain considered.

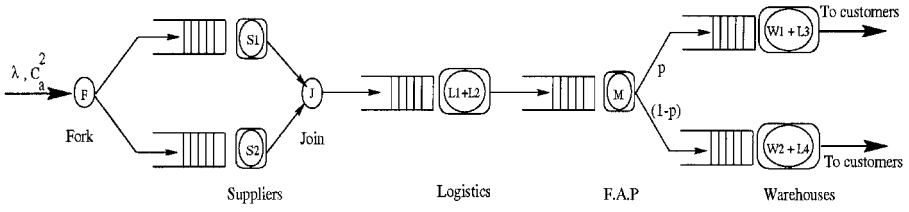

Figure 7. The supply chain network (FJQN) for manufacturer owned logistics.

nodes and the supplier times are obtained by the definition of SCVs to keep the comparison on par. To clarify, the SCV of $S_1 + L_1$ is given in table 10. We set the mean service times of S_1, etc. as in table 12. Assuming independence between the supplier service times and the logistics times, we can compute the SCV of L_1 once we fix the SCV of S_1, keeping in view that $S_1 + L_1$ is composed of S_1 and L_1. We thus can get the mean service time and SCV of $L_1 + L_2$. Also, all other nodes have service rates identical to those of figure 6.

For ease of exposition, we omit the comparison with simulations. From table 13, we see that the manufacturer owned logistics (MOL) seems to perform better than the supplier owned logistics (SOL) at high utilizations (or, high arrival rates, with service rates of the joining nodes remaining the same). This is with respect to the mean supply chain lead time for servicing the orders and the WIP inventory. On the other hand, at low arrival rates (consequently, low utilization of the joining nodes), we find that SOL performs better. Thus we see that the approximation developed above can be used in such crucial decisions as the ownership of logistics operations.

3.5.2. *Delivery performance and reliability*

Another application of the above approximation is in the calculation of delivery reliability which we define as the probability that customer desired due dates can be met. Let us assume that the total lead time is given as the maximum time taken to deliver any customer order (refer to figure 7). This time is essentially the time taken to serve a customer at warehouses W_1 *and* W_2. We note that the GQN analysis that we have done above can be intelligently used to buy us this value. Specifically, we can estimate the delivery time as the maximum of the waiting times incurred as one traverses until W_1 or W_2. For instance, if we consider the sub-network from the FAP onwards, we can write this as: $\max\{T_M + T_{W_1}, T_M + T_{W_2}\}$. In other words, $T = T_M + \max\{T_{W_1}, T_{W_2}\}$. One can easily evaluate the variance of the last stage

λ and C_a^2	1.0/h deterministic
μ_{S_1} and μ_{S_2}	3.6/h, 3.0/h
$C_{s_1}^2$ and $C_{s_2}^2$	(0.11, 0.11)
$\mu_{L_1+L_2}$	1.8/hr
$C_{l_1+l_2}^2$	0.1822
μ_M and C_M^2	3.0/h, 0.11
$\mu_{W_1+L_3}$ and $\mu_{W_2+L_3}$	2.0/h, 2.5/h
$C_{w_1+l_3}^2$ and $C_{w_2+l_4}^2$	0.11, 0.11

Table 12. Input parameters for the supply chain of figure 7.

λ (orders/h)	Mean supply chain lead time (h)		Average WIP	
	SOL	MOL	SOL	MOL
1.0	2.04	2.29	2.01	2.03
1.2	2.07	2.29	2.27	2.17
1.4	2.44	2.30	3.00	2.29

Table 13. Comparison of manufacturer owned and supplier owned logistics.

waiting times from the GQN analysis. We can compute T_{W_1} and T_{W_2} by their first two moments from the analysis of the GQN. Hence we can compute the first two moments of T by Clarke's method, of course, with the assumption that the constituents are normally distributed. Thus the probability that T is greater than a customer specified due date d can be computed using Chebyshev's inequality (Trivedi 1982), as:

$$\mathbb{P}(|T| \geq d) \leq \frac{\mathbb{E}[|T^2|]}{d^2}. \tag{8}$$

Thus, an upper bound on the delivery reliability can be obtained, which can be used by the process owner of the supply chain while deciding on whether to accept an order or not. From the above, we see how our method, which is simple to use, enables a quick analysis of supply chain decisions.

4. Conclusions

Performance modelling intended for decision making in supply chains is a critical issue. In this paper, we have presented queueing network based models for analysing supply chain networks in a dynamic and stochastic setting. In the first part of this paper we used generalized queueing models for computing the lead time and evaluated the influence of variability of lead times and demand. The first two examples illustrate the power of queueing network models in the variability analysis. We have specifically considered real world structures for supply chains. In the second part of this paper, we considered fork-join queueing systems and presented an approximate method for performance analysis. Through the use of two more examples, we illustrated the use of this method for decision making in supply chains.

References

ARNTZEN, B. C., BROWN, G. G. and HARRISON, T. P., 1995, Global supply chain management at Digital Equipment Corporation. *Interfaces*, **25**, 69–93.

BACCELLI, F., MAKOWSKI, W. A. and TOWSLEY, D., 1989, Acyclic fork-join queueing systems. *Journal of the Association of Computing Machinists*, **36**, 615–642.

BUZACOTT, J. A. and SHANTIKUMAR, G., 1993, *Queueing Models of Manufacturing Systems* (Englewood Cliffs, NJ: Prentice Hall).

CLARK, C. E., 1961, The greatest of a finite set of random variables *Operations Research*, March–April, 145–161.

CONNORS, D., AN, C., BUCKLEY, S., FEIGIN, G., LEVAS, A., NAYAK, N., PETRAKIAN, R. and SRINIVASAN, R., 1995, Dynamic modeling of re-engineering supply chains. Technical Report RC-19944, IBM Research Center, Almaden.

ERENGUC, S. S., SIMPSON, N. C. and VAKHARIA, A. J., 1999, Integrated production/distribution planning in supply chains: An invited review. *European Journal of Operations Research*, **115**, 219–236.

FLATTO, L. and HAHN, S., 1984, Two parallel queues created by arrivals with two demands: I. *SIAM Journal of Applied Mathematics*, **44**, 1041–1053.

FORRESTER, J. W., 1961, *Industrial Dynamics* (Cambridge, MA: MIT Press).

GERSHWIN, S. B., 1994, *Manufacturing Systems Engineering* (Englewood Cliffs, NJ: Prentice Hall).

KAMATH, M., 1997, RAQS: A software package for rapid analysis of queueing systems. Technical report, Centre for CIM, School of Industrial Engineering and Management, Oklahoma State University.

KIM, C. and AGRAWALA, A. K., 1989, Analysis of the fork-join queue. *IEEE Transactions on Computers*, **38**, 250–255.

KUMAR, A. and SHOREY, R., 1993, Performance analysis and scheduling of stochastic fork-join jobs in a multicomputer system. *IEEE Transactions on Parallel and Distributed Systems*, **4**, 1147–1154.

KUMAR, V. K., 1994, A package for performance analysis of generalized queueing networks. Technical report, Department of Computer Science and Automation, Indian Institute of Science, Bangalore.

LEE, H. L. and BILLINGTON, C., 1995, The evolution of supply-chain-management models and practice at Hewlett-Packard. *Interfaces*, **25**, 42–63.

MASCOLO, M. D., FREIN, Y. and DALLERY, Y., 1996, An analytical method for performance evaluation of Kanban controlled production systems. *Operations Research*, **44**, 50–64.

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

NGUYEN, V., 1993, Processing networks with parallel and sequential tasks: Heavy traffic analysis and Brownian limits. *The Annals of Applied Probability*, **3**, 28–55.

NGUYEN, V., 1994, The trouble with diversity: Fork-join networks with heterogeneous customer populations. *The Annals of Applied Probability*, **4**, 1–25.

SANKARAN, J. K. and SRINIVASA RAGHAVAN, N. R., 1997, Locating and sizing plants to bottle propane in South India. *Interfaces*, **27**, 1–15.

SILVER, A. E., PYKE, D. F. and PETERSON, R., 1998, *Inventory Management and Production Planning and Scheduling* (New York: Wiley).

SRINIVASA RAGHAVAN, N. R., 1998, *Performance Analysis and Scheduling of Manufacturing Supply Chain Networks*. PhD thesis, Indian Institute of Science, Bangalore.

SRINIVASA RAGHAVAN, N. R. and VISWANADHAM, N., 1999, Performance analysis of supply chain networks using Petri nets. *Thirty Eighth IEEE International Conference on Decision and Control*, Phoenix, Arizona, USA, December 1999.

TRIVEDI, K. S., 1982, *Probability and Statistics with Reliability, Queueing, and Computer Science Applications* (Englewood Cliffs, NJ: Prentice Hall).

VARMA, S. and MAKOWSKI, A., 1994, Interpolation approximations for symmetric fork-join queues. *Performance Evaluation*, **20**, 245–265.

VISWANADHAM, N. and NARAHARI, Y., 1992, *Performance Modeling of Automated Manufacturing Systems* (Englewood Cliffs, NJ: Prentice Hall).

VISWANADHAM, N. and SRINIVASA RAGHAVAN, N. R., 1999, Lead time models for analysis of supply chains. *Second AEGEAN International Conference on Analysis and Modeling of Manufacturing Systems*, Tinos Island, Greece, May 1999, pp. 325–334.

VOLLMAN, T. E., BERRY, W. L. and WHYBARK, D. C., 1998, *Manufacturing Planning and Control Systems*, fourth edition (The Dow Jones-Irwin/APICS Series in Production Management).

WHITT, W., 1983, The queueing network analyser. *Bell Systems Technical Journal*, **62**, 2779–2815.