Performance Modeling of Supply Chains using Queueing Networks

N. Viswanadham
Mechanical and Production Engg.
National Univ. of Singapore, Singapore 119260
e-mail: mpenv@nus.edu.sg

N. R. Srinivasa Raghavan
Management Studies
Indian Institute of Science, Bangalore, India 560 012
e-mail: raghavan@mgmt.iisc.ernet.in

Abstract

Supply chain networks are formed out of complex interactions amongst several companies whose aim is to produce and deliver goods to the customers at the time and place specified by them. 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 fork-join queueing networks to compute the mean and variance of the lead time. The existing literature on approximate methods of analysis of fork-join queueing systems assume heavy traffic and require tedious computations. We present two applications of a tractable approximate analytical method for lead time computations in a class of fork-join queueing systems. For the case where the arrivals are deterministic and service times are normally distributed, we present an easy to use approximate method. Specifically, we illustrate the use of the above method in setting service levels in assemble-to-order type supply chains.

1 Supply Chain Networks

Supply chain networks (SCNs) are formed out of complex interconnections amongst 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, 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 material 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 big role in the over all functioning of the SCP. In most cases, there is an integrator for the network, who could be an original equipment manufacturer, coordinating the flow of orders and materials through out the network. Modeling 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 modeling approach to compute 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 supply chain networks and the different order-fulfilment policies.

1.1 Operational Models

An important aspect of the supply chain operation is the supply chain planning and control methodology (SPC). 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: Make-to-stock (MTS), Make-to-order (MTO), Assemble-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 [12].

1.2 Modeling of Supply Chains

Supply chain networks are discrete event dynamical
systems (DEDS) in which the evolution of the system de-
PENDS on the complex interaction of the timing of various
discrete events such as the arrival of components at the sup-
plier, 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 last two decades, there has been a tremen-
dous amount of research interest in this area. There are
several classes of models that are useful in this context.
These models can be used for either qualitative or quanti-
tative analysis. Qualitative analysis yields results on sta-
bility and deadlock analysis. Quantitative methods, on the
other hand, highlight the determination of system perfor-
ance measures such as throughput and lead time. Series-
Parallel graphs, Petri nets and queuing networks are fund-
damental models for DEDS. Discrete event simulation is a
very general method and is widely followed. System dy-
namic models are also widely used for supply chain perfor-
ance evaluation [10].

1.3 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 cru-
cial 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 vari-
ance of the supply chain lead time can help one in quoting
reliably the delivery time 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 effi-
cient approximation algorithms for computing the expected
supply chain lead time and variance. This is the subject mat-
ter of Section 2. We remark that existing literature on ap-
proximate methods of analysis of fork-join queueing (FJQ)
systems assume heavy traffic [9] and require tedious com-
putations. For a class of fork-join queueing systems, we
present in Section 3, a new approximate method of analysis
which we use in the analysis of supply chains. For the case
of deterministic arrivals and normally distributed process-
ing times, we present an easy to use approximate method,
based on results of Clarke [3]. We report encouraging re-
sults for this class of fork-join systems, with some possible
applications in the supply chain context. We conclude this
paper in Section 4.

2 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. Consider the following supply chain:

- 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.

- This product (family) is manufactured from two ma-
  jor sub-assemblies supplied by two different suppli-
  ers. The inbound logistics is managed by the suppli-
  ers themselves.

- The sub-assemblies are then joined at a manufactur-
  ing plant. There is a synchronization delay 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 i.e. orders for the sub-assemblies
  are placed simultaneously with the suppliers. This is
  a structural feature that simplifies the analysis of the
  underlying FJQ system.

- The assembled end product (family) is then delivered
to distributors.

We model such a SCN by a queueing network as shown in
Figure 3. Observe that this queueing network has fork-join
structure preceding a generalized queueing network. Once
we analyze the fork-join structure, we can easily further the
analyze using well known approximations [2]. We are inter-
ested in computing the end-to-end delay, or the total mean
supply chain lead time. It is well known (see [5]) that FJQ
systems are difficult to analyse exactly. Hence many ap-
proximations have been proposed in the literature (see the
references in [6, 1]). Exact results are available only for the
case where the arrivals are Poisson, with exponential ser-
vice times and just two joining nodes. See [4, 8] for details.
Most of the approximate methods assume the following:

- The processing times at the servers belong to the ex-
  ponential family of distributions (exponential or Er-
langan 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 times 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 thrice the standard deviation and in real life supply chains, this assumption is found to be adequate and reasonable [7].

2.1 Computing the SCV of Inter-departure Time

Consider two servers with general service times and infinite buffer sizes as shown in Figure 2. 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 node is the arrival process to any downstream server. Once the moments of the above departure process are computed, the analysis of the remaining part of the queueing network is at hand. In Table 1 we detail the notation used. The average values will be denoted by $E[.]$. Under conditions of stability, 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. 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, 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 [2], pp-75. We ignore the effect of blocking of servers and make the (first) assumption that the mean inter departure rate is same as the mean inter arrival rate for each server. Thus, when there is a fork to the two servers,
Table 2: Input parameters (Case A) for the supply chain of Figure 3

<table>
<thead>
<tr>
<th>N</th>
<th>( \rho ) (%)</th>
<th>Mean flow time at FJ stage in seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Case A</strong></td>
<td><strong>Exact</strong></td>
<td><strong>Approximation</strong></td>
</tr>
<tr>
<td>2</td>
<td>50</td>
<td>382.07</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>385.26</td>
</tr>
<tr>
<td>10</td>
<td>50</td>
<td>460.80</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>474.49</td>
</tr>
<tr>
<td></td>
<td>90</td>
<td>461.62</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Case C</strong></th>
<th><strong>Case D</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>90</td>
</tr>
</tbody>
</table>

The supply chain network (FJQN) considered for study

\[ P_N, i=1..K, \text{Mean processing times (secs): 200 to 380 in steps of 20} \]
\[ \text{Standard deviation of processing times: } \frac{1}{2} \text{ of mean} \]
\[ \text{Squared coefficient of variation of all processing times: 0.0625} \]
\[ \text{Mean inter-arrival time seconds: 760, 475, and 422.2} \]

2.3 Numerical Results

For purposes of validation, we very briefly 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 2. In the table, the mean inter-arrival time is specified for varying utilization values of the server with the maximum service time of 380 seconds. The utilizations considered were 50, 80, and 90% respectively at the server with the maximum mean service time.

In this table, Case B refers to the case where the servers are identical with service times equal to 380 seconds, for both two and ten server systems. Case C refers to non-identical servers with same mean service times as Case A, but their standard deviations decrease from 50 seconds (for the server with mean service time 200s) to 5 seconds (for the server with mean service time 380s) in steps of 5s. Case D is similar to Case C, but the service time standard deviations now decrease from 150s to 60s in steps of 10s as in Case C. The results are tabulated in Table 3. The maximum absolute error percentage from the table shown is found to be 12%. We note that this occurs for Case B (with ten identical servers) and when the utilization value is 90%. Also, Clarke [3] showed that the approximation for the maximum value of a 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 identical 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 difference of the approximated flow time from that computed from simulation, divided by the latter.

2.4 Setting Service Levels in a Two Echelon Assemble-to-order System

The approximation that we have developed, can also aid in computing the total costs in certain assemble-to-order supply chains, in the process ‘determine’ the service levels, which is defined as 1-{probability of stockout}. The two main cost components in such supply chains are the inventory and the delay costs. We present a simple illustration for this case. Consider the supply chain in Figure 3 which shows a two echelon supply chain with suppliers at the first echelon and the OEM as the second. Now, let us assume that the system is operated in a assemble-to-order fashion, rather than the make-to-order type which was considered in the earlier discussion. This would mean that we will have inventories of components bought from suppliers S1 and S2, that are assembled at M and sold to end customers on order. In the ensuing discussion, we omit the warehouses from the analysis. Name the components bought from S1 and S2 as A and B respectively. Let C be the finished goods. Let us assume that the components are ordered when M is out of stock. We associate probabilities of stock outs for A and

\[
\mathbb{E}^2[D] = (\mathbb{E}[D_1]^2 + \sigma_{D_1}^2) \Phi(\alpha) + \frac{a}{a}[\mathbb{E}[D_1] - \mathbb{E}[D_2]] \Phi(-\alpha)
\]
\[
\alpha = \frac{1}{a}[\mathbb{E}[D_1] - \mathbb{E}[D_2]]
\]
\[
\sigma_D^2 = \sigma_{D_1}^2 + \sigma_{D_2}^2 - 2\sigma_{D_1} \sigma_{D_2} \rho
\]
B. Let us assume that the suppliers manage the inbound logistics, and that the processing times of individual components at these suppliers are known, along with the logistics times. We can thus aggregate these two along with any interfaces present, by the servers S1+L1 and S2+L2 of Figure 3. For ease of exposition, let us assume that the probabilities of stock outs of A and B are the same, and equal to \( p \). Also \( p^2 \) be the probability that M is out of stock of A and B simultaneously. This is usually a management set parameter, and hence a decision variable. (Our analysis is easily extendable to the case when A and B have different probabilities of stock outs.) The target inventories of A and B at M (respectively, \( I_A \) and \( I_B \)) are set based on the stock out probabilities. For instance, see [12] on how this can be done. This requires an assumption that the lead times from the suppliers (including processing at their factories, logistics and interface times) are normally distributed, to make matters tractable. Let \( D_1 \) and \( D_2 \) denote the waiting times from S1 and S2, respectively. Using mean and variance analysis one can determine the waiting time at M.

Thus the total average lead time for arriving orders is obtained as:

\[
D = p(D_1 + D_2 + p \cdot \mathbb{E}[\max(D_1, D_2)]) + D_3
\]

Similarly, the total average inventory in the system is given by:

\[
I = (1 - p)(I_A + I_B) + p(L_1 + L_2 + p(L_1 + L_2)) + I_3.
\]

where \( L_1 \), \( L_2 \), and \( L_3 \) are the steady state average WIP at S1, S2, and M respectively, which are computed using GI/G/1 analysis of the respective servers. Observe that \( \max(D_1, D_2) \) is at hand, thanks to Clarke’s method. All other values are calculable using simple approximate solutions of GI/G/1 queues. Hence one can, in principle, perform a total cost analysis as follows. Let \( H_1 \) be the holding costs of inventory and \( H_2 \) the delay costs. Thus the total cost of operating the supply chain in the assemble-to-order fashion, is given as

\[
TC = H_1 \cdot I + H_2 \cdot D
\]

For varying values of \( \frac{H_2}{H_1} \), one can then compute the total cost, thus enabling the setting of the stock out probability \( p \), in turn, the target inventories of A and B at M. We note here that, although \( H_1 \) is assumed to be the same for inventories of the components and the finished goods, in practice, it is possible to have the holding costs of finished goods to be, say, 20% higher than those of the components. Similarly for the delay costs \( H_2 \). This can be easily incorporated into our analysis by altering the total cost function suitably, although we do not do that here. The input case considered is shown in Table 4. As discussed above, the various variables are computed for the given input parameters, and we get the following expression for the total cost:

\[
TC = H_1[0.370 + (1 - p)(I_A + I_B) + 2.169p] + 2.169p^2 + H_2[0.037 + 0.217p + 0.174p^2]
\]

(10)

The above equation can be used to determine the total cost given \( p \). Alternatively, if \( p \) is a decision variable, we enumerate for various values of \( p \) and get the least cost solution. We know that

\[
p = \mathbb{P}(DDL_i \geq t_i), i = A, B
\]

where DDL is the demand during lead time (i.e., the orders for finished goods that arrived even when the required components are on order) which is a random variable. This is obtained as the product of the arrival rate and the lead time for replenishment. Assuming now, that each arriving order for finished goods C requires one component each of A and B, we compute the following:

\[
DDL_i = \lambda * D_i, i = A, B
\]

(12)

\[
C_{DDL_i}^2 = C_{D_i}^2, i = A, B
\]

(13)

We now make the assumption that the DDLT computed above, is a Gaussian random variable. Using the definition of \( \lambda \), we can easily determine \( I_A \) and \( I_B \). For various values of the stock out probability \( p \), and the ratio of \( \frac{H_2}{H_1} \), we computed the total costs, the same being presented in Figures 4–5. The trend shown in the graphs is expected, because, as the probability of stock outs is allowed to decrease, the inventories go up and vice versa. The minimal total cost can thus be traced to an appropriate value for \( p \), although it requires exhaustive enumeration.

### 3 Conclusions

Performance modeling 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.

| \( \lambda \) and \( C_{D_i}^2 \) | 10/day, 0.5 |
| \( \mu_{S_1} \) and \( \mu_{S_2} \) | 15/day, 20/day |
| \( C_{D_1}^2 \) and \( C_{D_2}^2 \) | (0.8, 0.4) |
| \( \mu_{S_3} \) and \( C_{D_3}^2 \) | 30/day, 0.2 |

**Table 4:** Input parameters for the assemble-to-order supply chain.
Specifically, we considered fork-join queueing systems and presented an approximate method for performance analysis. We presented a potential application of this method in setting service levels in certain assemble-to-order supply chains.

References


