Journal of the Operational Research Society (2000) 51, 1158–1169
www.stockton-press.co.uk/jors

Performance analysis and design of supply chains: a Petri net approach

N Viswanadham[1] and NR Srinivasa Raghavan[2]*

[1]*National University of Singapore and* [2]*Indian Institute of Science*

In this paper, we investigated a dynamic modelling technique for analysing supply chain networks using generalised stochastic Petri nets (GSPNs). The customer order arrival process is assumed to be Poisson and the service processes at the various facilities of the supply chain are assumed to be exponential. Our model takes into account both the procurement process and delivery logistics that exist between any two members of the supply chain. We compare the performance of two production planning and control policies, the make-to-stock and the assemble-to-order systems in terms of total cost which is the sum of inventory carrying cost and cost incurred due to delayed deliveries. We formulate and solve the decoupling point location problem in supply chains as a total relevant cost (sum of inventory carrying cost and the delay costs) minimisation problem. We use the framework of integrated GSPN-queuing network modelling—with the GSPN at the higher level and a generalised queuing network at the lower level—to solve the decoupling point location problem.

Keywords: supply chain management; stochastic Petri nets; performance analysis

The supply chain networks

Manufacturing 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, warehouses, distributors, retailers and customers (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, planning, distribution, manufacturing and purchasing activities independently, optimising their own functional objectives. SCM is a process-oriented approach to coordinating these processes across all organisations and all functions involved in the value delivery process.

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 discrete event dynamic system and present a Petri net[1] based modelling approach to compute performance measures such as lead time and work in process inventory. In particular, we investigated the use of generalised stochastic Petri net models for comparing make-to-stock and assemble-to-order policies in supply chains in terms of total cost, which is the sum of inventory and delay costs.

In global manufacturing industries, it is not uncommon to find a part of the supply chain operating under supply push conditions and the other part under demand pull conditions. The point at which this transition takes place is called the decoupling point. Locating the order decoupling point along a supply chain is an important decision.[2] In this paper, we present an approach to find the decoupling point, based on integrated queuing-Petri net representation of the SCN. We represent the push and pull portions of the SCN separately as queuing networks and the integrated SCN using a a higher level Petri net. After solving the resulting sub networks, we use an enumeration based algorithm for locating the decoupling point so that inventory costs are traded off with customer order delay related costs.

Supply chain network configuration

The configuration of the supply chain defines the interconnection pattern among its facilities. Supplier's manufacturing plants, logistics, final assembly plants, packaging centres and transshipment points (cross docking) are examples of facilities. All supply chain networks do not have the same configuration. Depending on the product structure several network configurations are possible. One could identify four major supply chain configurations including serial, converging, diverging, and converging-diverging networks. The supply chains can be operated in different

**Correspondence:* *Dr NR Srinivasa Raghavan, Management Studies, Indian Institute of Science, Banglore, India 560 012.*
E-mail: raghavan@mgmt.iis.ernet.in

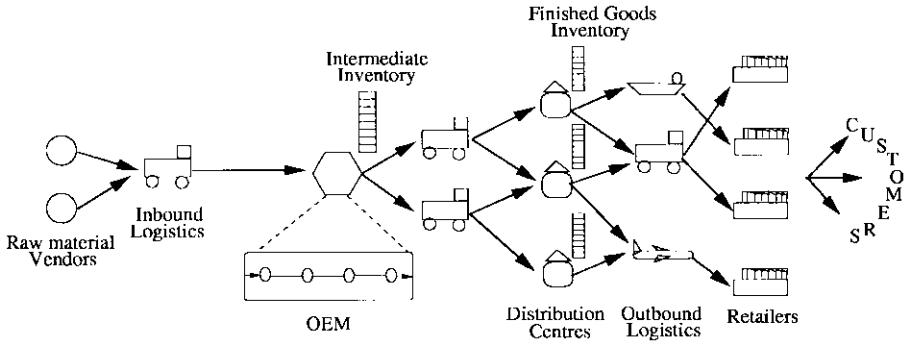

Figure 1 The supply chain network.

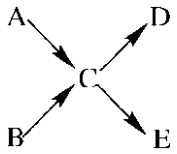

Figure 2 Product structure for supply chain considered.

ways: make-to-stock, make-to-order or assemble-to-order. We briefly cover these issues here.

1. *Serial structure*: Here, one facility of the network feeds into another and the entire supply chain resembles a single pipe line.
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. See Figure 2 for example.

In this paper, we concentrate on the serial structure. The application of the methodology to other structures is an easy extension.

Operational models

Another important aspect of of supply chain operations 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 synchronised 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 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. This qualifies for the supply chain design problem.

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 Reference 3. In this paper, we present analytical models that aid in determining as to when to prefer an assemble to order policy and when, the traditional make to stock policy.

Literature survey

In this section, we briefly survey the literature on mathematical models for supply chains. The analytical modelling

of supply chain networks can broadly be classified 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 literature on this subject and a number of survey papers also appeared on this subject (see Reference 4 and the 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*.[5]

Performance analysis of SCNs is basically conducted to determine the lead time, variation, cost, reliability and flexibility. SCNs are discrete event dynamical systems (DEDS) 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 last 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, to name a few. 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.

Analytical models

Our aim here is to summarise five analytical techniques useful for modelling SCNs including series-parallel graphs, Markov chains, queuing networks, Petri nets and system dynamics models.

Series parallel graphs. Series parallel graphs can model an SCN by assigning probability distributions to the lead time of the activities in the graphs. These are graphs, showing 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.

Markov chains. The use of Markov models in the study of performance of manufacturing systems[6] is well known. Direct modelling of an SCN as a Markov chain would be very difficult and expensive. Higher level models based on Petri nets and queuing networks are ultimately solved as Markov chains using software packages such as SPNP.

Petri nets. Faithful modelling of iteration, synchronisation, forks, and joins that arise in SCNs is possible using Petri nets. If too detailed models are developed, numerical solution, however, may turn out to be a nightmare. Hierarchical modelling discussed in this paper provides a tractable way of handling largeness here.

Queuing networks. The most general SCN can be modeled as a fork-join queuing network model with iteration or reentrancy. An analytical solution of these general models is not available, and approximations are available in only special cases, some solutions can be found.[7] This is an area of active research.

System dynamics models. Here the SCN is model using differential equations. Forrester first explained the bullwhip effect using these models.[8]

We consider the modelling of SCNs using Petri nets in the following.

Performance analysis of SCNs

In this paper, we abstract the supply chain at the organisation level as shown in Figure 1. The information required for such modelling, like the processing times at the various facilities, are assumed to be available *a priori*. We wish to study the dynamics of the supply chain, especially, the impact of logistics and interfaces, and the manufacturing philosophies, on the performance measures that we will define later. Such an aggregate level analysis is common in the factory-floor literature.[9]

Introduction to Petri nets

We use the framework of generalised stochastic Petri nets (GSPN) for the analysis. It is now a widely used and well accepted methodology for the analysis of manufacturing systems. The interested reader may find a detailed treatment of the same in Reference 6.

A GSPN is a 8-tuple ($P, T, IN, OUT, INH, M_0, F, S$) where P is a finite set of places; T is a finite set of transitions (P and T are disjoint); Transitions can be two types: immediate or exponential (timed), depending on whether they fire in zero time or random times; IN and OUT are the set of input and output functions that define directed arcs from places to transitions and vice versa; INH is the inhibitor function and defines the corresponding arcs (the presence of a token in the input places of these arcs will result in the transition being inhibited from firing); M_0 is the initial marking; F is a firing function associated with each exponential timed transition in each marking; S is a set, possible empty, of random switches associating probability distributions to subsets of conflicting immediate transitions.

Pictorially, places are represented by circles and transitions by bars. Places in a Petri net usually represent conditions or resources in the system while transitions

model activities. Dynamic analysis of GSPNs involves generating the reachability tree, also called the marking process, starting from an initial marking. This process is automated and software packages such as stochastic petri net package (SPNP) are available. It is known that the marking process of a GSPN is a semi-Markov process with a discrete state space, given by the reachability set of the GSPN. Various performance measures like lead times, inventories and the throughput can be computed from the stationary probabilities of the reduced embedded Markov chain. We use the SPNP also for purposes of numerical analysis[10] of the underlying semi-Markov process.

Performance analysis illustrated

Consider the supply chain network shown in Figure 3, with two suppliers for two sub-assemblies, an OEM assembling these, and two separate logistics providers to handle the transport of the sub-assemblies to the OEM. Two distribution centres are the end customers for the products. We model this supply chain network under two different policies for material flow, viz., make-to-stock and assemble-to-order systems. The inter-arrival time of orders for the end products are assumed to be exponentially distributed. All processing and transporting times are also assumed to be exponentially distributed. More general distributions can be handled[11] but at the expense of much numerical complexity. Since our purpose here is only demonstration of the technique, we will proceed with the exponential assumptions.

The product structure considered for study is shown in Figure 2. There are two end products D and E which are available at warehouses W_1 and W_2 respectively. The demand for these two products are stochastic. The end products D and E are assembled at the final assembly plant or OEM, namely, *M*. The common base sub-assembly for D and E is C. Also, C is assembled from raw materials A and B provided by suppliers S_1 and S_2 respectively. Inventories of A, B, C, D, and E are maintained at the respective facilities.

We model the dynamics of the above SCN shown in Figure 3 by using Petri nets. Our modelling is more generic and wholesome than current approaches in the following sense. We consider the logistics process also in the study.

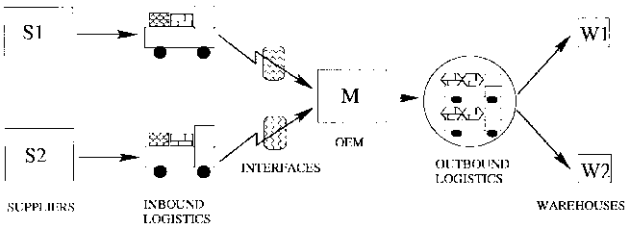

Figure 3 The supply chain considered for Petri net modelling.

Logistics is typically given a cursory treatment in available literature,[12] by assigning constant values for the transportation times. We allow random logistics times here in our study. Also, crucial issues like interfaces discussed earlier in this paper, are not usually considered in current literature. We assume that the average times spent at the interface servers are known, along with their variances. Since we model such a network using GSPNs, we consider exponential service times alone. We note that while the inbound logistics into *M*, is managed by the suppliers individually, the outbound logistics out of *M* is managed by the OEM. This is brought out in Figure 3 by having a common pool of logistics carriers that move products out of *M*.

We consider continuous review reorder point inventory control policy. For ease of analysis, we assume that each arriving order for D or E, is for a *batch* rather than single items. Stock piles of D and E are replenished at fixed reorder points, which are preset.

The Petri net for the above case is shown in Figure 4. The description of the GSPN is given in Tables 1 and 2. We define enabling functions for the transitions *tA*, *tB*, *tC*, *tD*, and *tE*. These immediate transitions are enabled only when the tokens in the places representing inventory for A, B, C, D, and E reach their respective reorder points. Observe that once this condition is satisfied, the transitions can keep on firing indefinitely. To avoid this, we define inhibitor arcs from places *PA*′, *PB*′, *PC*′, *PD*′, and *PE*′, to the above transitions. Therefore these places signify that material is already on order. The tokens from the above places are removed once the material is available for transporting to the respective inventory locations, which occurs when there is a token in the places *P*5, *P*6, *P*9, *P*13, and *P*14. The initial marking (as shown in the GSPN) consists of tokens in places *P*7, *P*8, *P*10, *P*16, and *P*17, equalling in number to the respective targeted finished goods inventory of A, B, C, D, and E. The SPNP package generates the reachability tree, the set of all markings, formulates the reduced embedded Markov chain and solves it for steady state probabilities. Now, one can find the statistics such as the expected number of tokens in a particular place, mean number of firings of an particular transition and mean waiting time of a token in a particular place. These can be interpreted as work-in-process inventory or lead-time depending on the context.

We add an interesting dimension to the above model. Instead of making the final products D and E to stock, what would happen if they were assembled to order, from the common base component C? The make-to-stock (MTS) kind of system offers better serviceability in terms of faster access to end products D and E, therefore reducing the probability of back ordering them. This naturally implies holding excess finished goods inventory which may get obsolete, if customer demands are not steady and dense in nature. The assemble-to-order case offers lower costs for the supply chain in terms of holding inventory, but at the

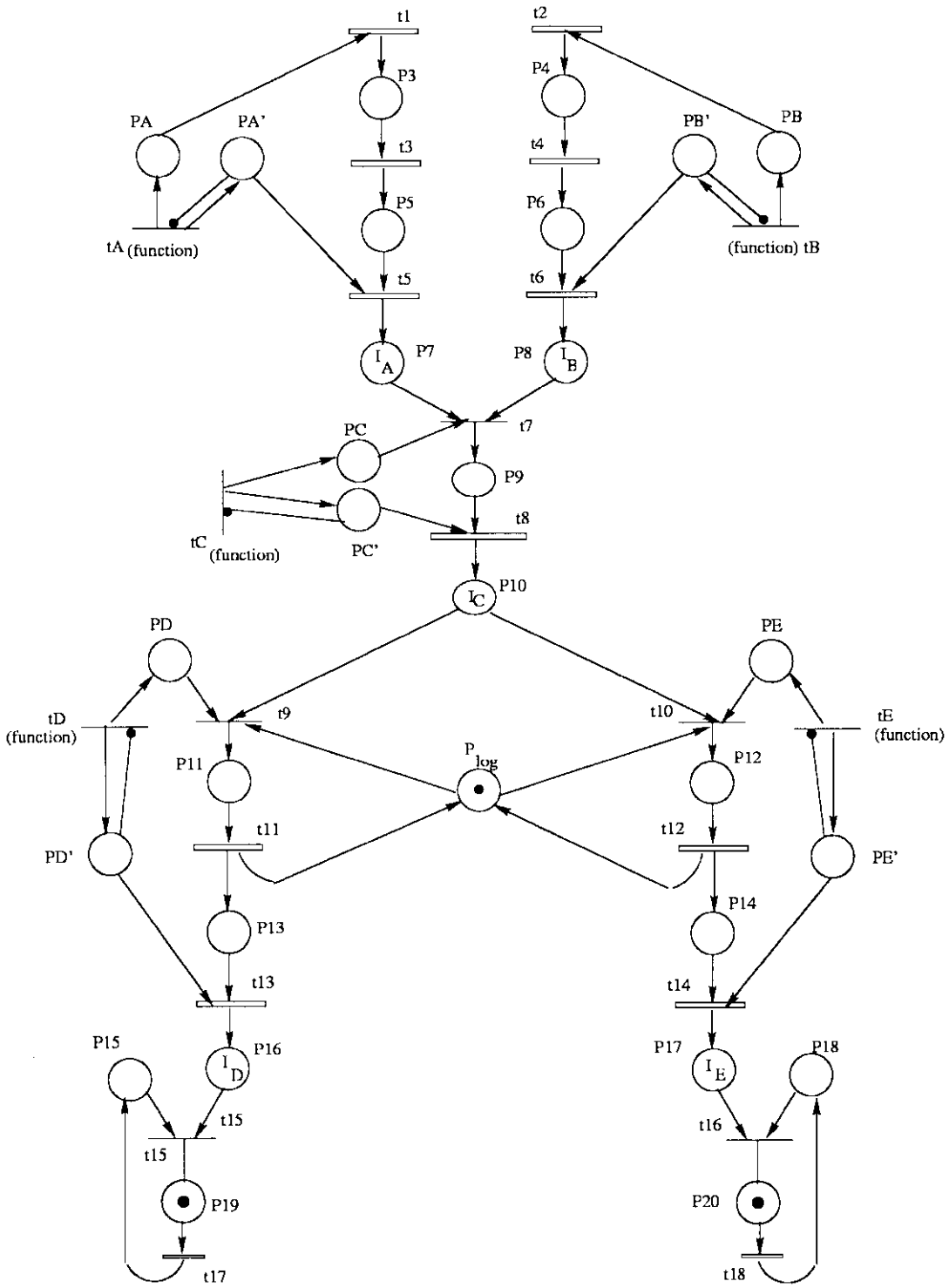

Figure 4 Petri net model for the supply chain with reorder point control. (See Tables 1 and 2 for keys).

expense of large customer lead times. That is to say that the orders from customers at warehouses W_1 and W_2 who cannot wait until the product is assembled to order, will be lost. We can modify the GSPN model for the assemble-to-order (ATO) case (which is not shown in the paper) when the production of D and E are triggered directly by customer orders. Hence the places PD, PD′, PE, and PE′ found in Figure 4 will be absent. Also transitions *tD* and *tE* are absent. The initial marking will not have tokens in places *P*16 and *P*17. The performance measures of interest are the

Table 1 Interpretation of places in the Petri net in Figure 4

Place name	*Description*
PA′	Material on order to supplier of A
PB′	Material on order to supplier of B
PA	Manufacturing at supplier of A
PB	Manufacturing at supplier of B
P3	Logistics from supplier of A
P4	Logistics from supplier of B
P5	Interface between supplier A logistics and OEM
P6	Interface between supplier B logistics and OEM
P7	Available inventory of A
P8	Available inventory of B
PC	Order receipt for production of C
PC′	Material on order for production of C
P9	Production of C
P10	Inventory of C available
PD	Order receipt for production of D
PD′	Material on order for production of D
PE	Order receipt for production of E
PE′	Material on order for production of E
P11	Outbound logistics of D from plant to warehouse
P12	Outbound logistics of E from plant to warehouse
Plog	Logistics carriers available
P13	Assembling of D from inventory of C
P14	Assembling of E from inventory of C
P16	Finished goods inventory of D at warehouse
P17	Finished goods inventory of E at warehouse
P15	Back order for D ready
P18	Back order for E ready
P19	Customer order for D ready
P20	Customer order for E ready

Table 2 Interpretation of transitions in the Petri net in Figure 4

Transition name	*Description*
tA	Start of manufacturing of A
tB	Start of manufacturing of B
t1	Processing by supplier of A
t2	Processing by supplier of B
t3	Transportation from supplier of A
t4	Transportation from supplier of B
t5	Paper work or interfaces with supplier of A
t6	Paper work or interfaces with supplier of B
tC	Trigger for production of C
t7	Manufacturer of C starts production
t8	Processing of C
tD	Trigger for assembling of D
tE	Trigger for assembling of E
t9	End of assembling of D from C
t10	End of assembling of E from C
t11	Outbound logistics of D
t12	Outbound logistics of E
t13	Assembling of D
t14	Assembling of E
t15	Customer order for D served
t16	Customer order for E served
t17	Arrival of order for D
t18	Arrival of order for E

average work in process and average finished goods inventories of materials/goods A through E; the lead times for order delivery; and, material replenishment cycle times (or, the supply chain lead times) for D and E. We define below a single objective function called total cost, which summarizes all the above measures into one.

Numerical results

Here, we wish to evaluate the performance of the supply chain in terms of the total cost, which is the sum of the total inventory carrying cost and the cost incurred due to delayed deliveries. It is clear that the above components of the total cost are complementary to each other. There is a trade-off between the two which is clearly brought out by the two replenishment policies followed, *viz.* the reorder point based make-to-stock system and the assemble-to-order system.

Let the holding cost incurred for inventories of the first stage (products A and B) be H_I, and the cost per hour of delayed delivery be H_D. We vary the ratio of the component costs, H_D/H_I from 1.5 to 40.0 to observe any trends in the total cost, so that the decision maker can choose the appropriate policy. We assume that product E is valued more than product D by 50%. That is, delayed deliveries for E are costlier by 50% when compared to that of D. Also, keeping in view that held inventory becomes expensive as we move from raw materials to finished goods, we increase the holding cost rates from the raw materials to the finished goods. Specifically, the finished good inventory of C is 20% costlier than that of A and B, and the finished goods inventories of D and E both cost 20% more than that of C.

We define the average net inventory as the total steady state work-in-process inventory and the finished goods inventory present in the supply chain. The net inventory is computed as follows:

- For the make-to-stock system, it is the sum of the steady state average finished goods inventories of A, B, C, D, and E.
- For the assemble-to-order system, it is the sum of the steady state average finished goods inventories of A, B, C, and the steady state average work in process inventories of D and E, since the last two are *not* made to stock.

The SPNP package gives in its output, the average number of tokens in the all places of the GSPN. The average net inventory is easily obtained from this information. Also, the average net delay in delivery of D and E is the sum of the steady state average customer order lead times for D and E, which is again obtained from the GSPN analysis. This is obtained from the SPNP output as follows. Let us consider computing the average delay in delivering, say, D. We need to know the steady state average number of products of type D 'leaving' the system, and the steady state average rate at which the product 'leaves' the system. The ratio of the above

two is the steady state average waiting time for that product. In this case (D), we compute the steady state average number of tokens in place $P16$ and the steady state throughput rate of transition t17 and take the ratio in that order to get the average delay for D. It is now clear that using SPNP we can obtain the total cost for a given supply chain configuration, and inventory control and SPC policies. We show the input parameters for the base case in Table 3. The targeted finished goods inventory of A, B, and C are 6 units while that of D and E are 3 units for the base case. The reorder points (inventory level that triggers reorder) of A, B, D, and E are 1 unit each while that of C is 10 units. We assume one outbound logistics vehicle for the base case. While the above concern the MTS case, for the ATO case, all the above values remain but for the fact that there are no more targeted finished goods inventory and reorder points for D and E which are the end products.

Effect of arrival rates of end products. The influence of arrival rates of the end products D and E on the performance of the system is shown in Table 4. We observe the following:

1. When the ratio $H_D/H_I = 1.5$, the assemble-to-order system dominates make-to-stock system. This implies that when delayed deliveries are not costly, it is better to have the assemble-to-order system. The total delay in delivery decreases as the arrival rate increases in the assemble-to-order system. This is a direct consequence of Little's Law which states that *Inventory* $= \lambda \times$ *Waiting Time*. Here, *inventory* is to be considered as the constant number of tokens floating in the assemble-to-order portion of the Petri net. On the other hand, the waiting times of the end products in the make-to-stock system were found to increase as their arrival rate increases. This is because the arriving orders for D and E are served from the finished goods stock pile. Hence as the arrival rate is increased, orders will have to wait until the stock pile is non-empty.
2. When the ratio $H_D/H_I = 40.0$, delayed deliveries are expensive. Therefore one would expect the assemble-to-order system to perform badly, and so it does, as in Table 4. Interestingly, in the case of the make-to-stock system, the total cost appears to have a U shape with the 'minimum' somewhere between arrival rates of 1.0 and 1.2. Therefore, it is more economical for the enterprise to go in for the make-to-stock system when the ratio is 40.0, and also, if the arrival rate can be managed to somewhere around 1.0 to 1.2 units/h, it will yield the best benefits.

Table 3 Transition firing rates for the Petri net in Figure 4

Transition name	*Firing rate (units/h)*
t1	1.00
t2	1.00
t3	3.00
t4	2.00
t5	6.00
t6	4.00
t8	2.00
t11	4.00
t12	2.00
t13	4.00
t14	3.00
t17	0.80
t18	0.60

Table 4 Variation of total cost with arrival rates of D

	Total cost			
	$H_D/H_I = 1.5$		$H_D/H_I = 40.0$	
λ_D *units/h*	*MTS system*	*ATO system*	*MTS system*	*ATO system*
0.8	22.421	19.815	26.001	257.437
1.0	21.237	18.610	25.818	237.559
1.2	20.012	17.714	25.961	224.228
1.4	18.774	17.016	26.339	214.675

Effect of targeted finished goods inventory of C. We studied the effect on the total cost of the targeted finished goods inventory of the common base material (C) for producing the end products D and E.

1. In Table 5 we show the results for the make-to-stock (MTS) system. As we increase the finished goods inventory (FGI) of C, the customer order delays decrease due to increased stock piles (and hence lower back order times). When the ratio of costs is 1.5, the inventories are taxed more than the delays, and hence we see that there is a 134% increase in total costs when the finished goods inventory of C is increased by 150%. On the other hand, when the delays are costlier and consequently inventories are taxed less (cost ratio = 40.0), we see that the total cost increases by about 78% only. This suggests that make-to-stock systems are preferable when delay related costs are substantial when compared to inventory costs. This corroborates available literature (see for instance Reference 13 on such systems.
2. In Table 6 we capture the results for the assemble-to-order (ATO) system. We see that, as in the case of the make-to-stock system, increasing the finished goods

Table 5 Variation of total cost with inventory of C in a MTS system

	Total cost	
FGI_C	$H_D/H_I = 1.5$	$H_D/H_I = 40.0$
6	18.54	28.01
9	27.53	29.34
12	35.553	42.175
15	43.403	49.929

Table 6 Variation of total cost with inventory of C in a ATO system

	Total cost	
FGI_C	$H_D/H_I = 1.5$	$H_D/H_I = 40.0$
5	15.64	197.40
6	18.37	201.52
7	21.07	204.87
8	23.73	207.92

inventory level of C increases the total cost. When the finished goods inventory of C is 6, the assemble-to-order system performs better than make-to-stock system, when the cost ratio is 1.5 (shown in first columns of Tables 5 and 6). Also, in the case of the assemble-to-order system, when the cost ratio is 40.0, the total cost increases by about 6% when the base stock levels are increased by 60% from 5–8. This gives us an indication that the assemble-to-order system is almost immune to the finished goods inventory levels of C when the cost ratio is high. But this is no incentive in going for such a system, for, though the percentage variation is small, the *absolute* values of total costs are far higher than is the case for the make-to-stock system (shown in the last columns of Tables 5 and 6.)

Effect of interface times of supplier of sub-assembly B. We now analyse the impact of the interface times on the performance of the make-to-stock and the assemble-to-order systems. We vary the interface times with the supplier of sub-assembly B alone as shown in Table 7. We observe the following: all other parameters remaining the same, when we increase the interface rates, that is, when we decrease the interface times with supplier of B, the total costs *increase*, marginally though. This is irrespective of the policy used. This seems to be counter intuitive, especially considering the fact that interfaces are deemed to be non-value adding activities (NVA). The explanation for this anomalous behavior is that when we decrease the interface times with the supplier of B, the inventory held at various locations increases, since more inventory is added at a greater pace. We found that the inventory of C increases, causing more holding costs. Strangely, we also found a slight increase in the delay times, causing an increase in the total cost. The lesson from this analysis, is then, that any attempts at eliminating the NVAs should be done keeping the global view of the supply chain. That is to say, the interface times of the supplier of sub-assembly A also has to be reduced *simultaneously* so as to derive benefits.

Table 7 Variation of total cost with interface rates of supplier of B

	Total cost			
Interface rates with	$H_D/H_I = 1.5$		$H_D/H_I = 40.0$	
S_2 *units/h*	*MTS system*	*ATO system*	*MTS system*	*ATO system*
4.0	22.566	15.542	24.934	197.185
5.0	22.651	15.640	24.981	197.360
6.0	22.709	15.705	25.038	197.502
8.0	22.780	15.785	25.109	197.659

The postponement problem

In this section, we present an integrated GSPN-queuing model to solve the decoupling point location problem (DPLP). The decoupling point[2] is the point in space, that is, the facility, where the order is assigned to the customer. In other words, up to the decoupling point, all the sub-assemblies are made to stock. After the decoupling point, the item is assigned to the customer and made-to-order. We consider a supply chain with a pipeline (tandem) structure as shown in Figure 5. Each upstream facility passes on material to the downstream facility. The end customer demands occur at the retail outlets, which are supplied by a single distribution centre. We consider only one type of end product. In such a supply chain, we would like to determine the customer order decoupling point. Our method can be easily extended to the multiple product types case provided that for each product type, from the decoupling point onwards, the SCN has a tandem structure.

Our approach is different by several accounts from the earlier studies.[2,14,15]

By taking recourse to integrated queuing Petri net models, we consider:

- The entire supply chain at an aggregate level, instead of only *stages* of manufacturing.
- Delays that occur at each of the facilities.
- The logistics process as a separate entity with random service times.
- The interface between the organisations.

We use an integrated queuing-GSPN model, with the GSPN model at the aggregated level and queuing model used to get the parameters for the GSPN. Therefore, the representational power of the GSPNs is combined with the algorithms available to solve general queuing models. Due

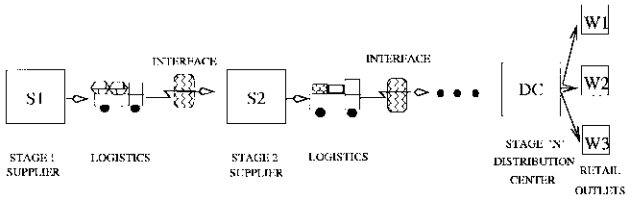

Figure 5 A pipeline supply chain structure considered for DPLP.

to the above, we can allow for general service times to solve the queuing model, while the resulting service rates are assumed to be exponential in order to solve the GSPN. This is reasonable, considering that the original GSPN model for the entire system would demand that all service rates be exponential. Instead of taking recourse to simulations[14] or approximations,[15] we approach the problem using an enumerative analysis of the underlying GSPN model. This is not costly, especially since we are interested in an aggregate average picture of the effect of the location decision.

An integrated GSPN-queuing network approach

The pipeline supply chain with N members under study here is shown in Figure 5. We illustrate the GSPN model for a two product type SCN in Figure 6. The single product type case is similar to the one shown in the above figure but for the fact that there will be only one subnet in the assemble-to-order portion of the supply chain. We present the interpretation of places and transitions for the single product type case in Table 8. The description for the two product type GSPN is similar to that in Table 8 but for the fact that the subnets corresponding to product types 1 and 2 have place and transition names suffixed with the product type; for example, P_{K1}, P_{bo2}, etc. We omit further details for the two product case from this paper.

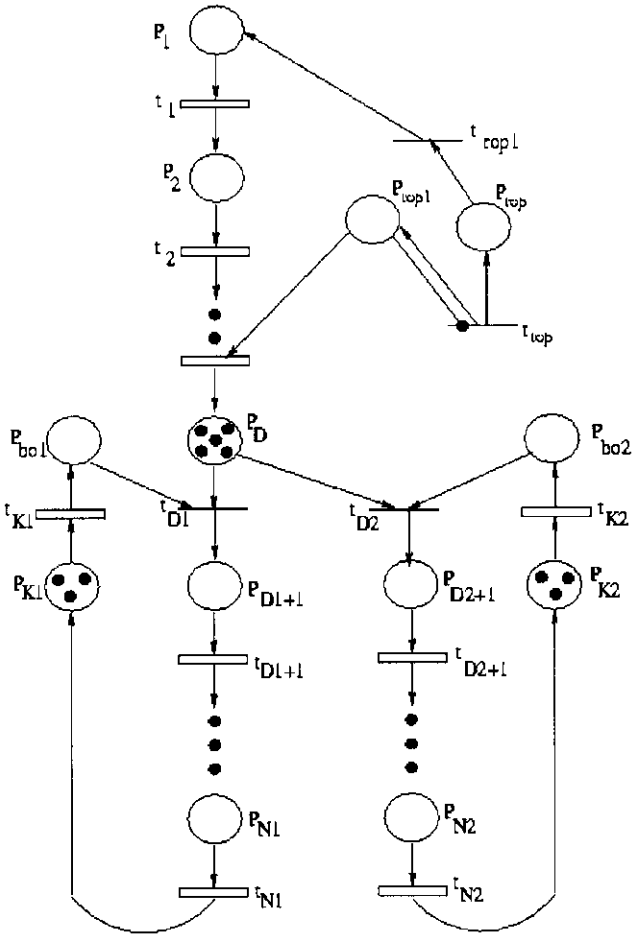

Figure 6 The GSPN model of a two product type supply chain. (See Table 8 for key).

For the single product type case, our aim is to determine the decoupling point. The costs considered are:

- The cost of holding inventory that results due to the make-to-stock part of the network, before the decoupling point D, and
- The cost of excess lead time required to assemble and deliver the customer order, from the decoupling point onwards.

The number of tokens in place P_D represents the targeted finished goods at the decoupling point. This is an input parameter, which is based on the desired customer satisfaction levels. This targeted inventory is replenished as soon as it touches a reorder point. In the remaining half (places $P_{D+1}, \ldots, P_N$) of the GSPN, Kanbans are used to trigger assembling. We observe that the above problem has a structure that makes it amenable for integrated queuing network-GSPN analysis (see Reference 6, pp 560–570). Specifically, we proceed as follows.

If we aggregate all the nodes from P_1 to P_D, the service rate of the aggregated facility can be obtained by solving the original product form queuing network (PFQN). The number of constant jobs circulating in the aggregated facility is equal to the targeted finished goods inventory at the decoupling point. Similarly, we aggregate the facil-

Table 8 Description of the places and transitions for the GSPN of the single product type SCN

Place name	*Description*
P_1	Orders in manufacturing plant at supplier 1
P_2	Orders in logistics from supplier 1
P_D	Inventory at decoupling point D available
P_{D+1}	Orders being assembled at facility D + 1
P_N	Orders in final customisation plant at distributor
P_K	Kanbans for end product available
P_{bo}	Orders waiting for inventory at D
P_{rop}	Order receipt for production at supplier 1, based on reorder point
P_{rop1}	Order already in process in the make-to-stock portion
Transition name	*Description*
t1	Processing by supplier 1
t2	Transportation from supplier 1
t3	Transportation from supplier of A
t_D	Trigger for assembly at D
t_{D+1}	Assembling at D
t_N	Final customisation at distributor
t_K	Arrival of orders for end product
t_{rop}	Trigger for production to supplier 1

ities in the assemble-to-order portion too, by solving another PFQN, this time with the number of constant jobs circulating equal to the number of Kanbans. The resulting throughput rates are given as the firing rates for the transitions t_1 and t_N shown in Figure 7. The firing rate of t_1 so obtained is, indeed, marking independent. This is because production in the make-to-stock portion is triggered only when the reorder point is reached. On the other hand, the firing rate of t_N is marking dependent, since the number of Kanbans floating is variable in the assemble-to-order portion. So, the PFQN for the assemble-to-order portion is solved K times where K is the number of Kanbans which again is an input parameter. The resulting GSPN is shown in Figure 7. Description of the places and transitions is given in Table 9.

We solve the DPLP using the above model. Our approach is enumerative in nature. That is, we analyse the underlying GSPN for all values of D ($=1,\ldots,N$). The total cost is obtained as the sum of aforementioned costs, and the minimum cost solution is chosen.

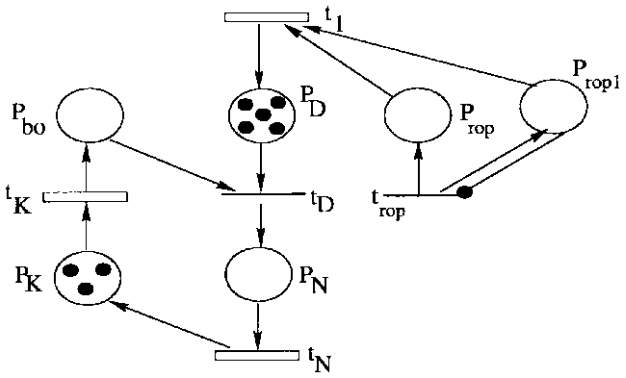

Figure 7 The aggregated GSPN model of the supply chain considered for DPLP.

Table 9 Description of the GSPN of Figure 7

Place name	*Description*
Prop	Material already on order to supplier 1
P_D	Inventory at decoupling point D
P_N	Orders in customisation at the aggregated facility
P_K	Kanbans for end product available
P_{bo}	Orders waiting for inventory at D
P_{rop}	Order receipt for production at supplier 1, based on reorder point
P_{rop1}	Order already in process in the make-to-stock portion
Transition name	*Description*
t1	Processing by aggregate facility
t_D	Trigger for assembly at D
t_N	Final customisation at aggregated facility
t_K	Arrival of orders for end product
t_{rop}	Trigger for production to the aggregated facility

When the service times are generally distributed, we analyse the underlying generalised queuing network (which is non product form type) by Whitt's approximations[16] and finally, while solving the higher level GSPN, we assume that the resulting service rates of the transitions t_1 and t_N are exponential. This is done so as to facilitate the use of existing computational techniques for solving GSPNs. Another way of solving the resulting GSPN is to use simulation.

Numerical results

In order to compute the component costs, we proceed as follows: (for notations we refer to Figure 7)

1. *Holding cost*: We assume that the holding costs increase as we move from the first stage suppliers to the distribution centre, by 20% for each stage. Hence the net inventory holding cost rate at the decoupling point is obtained as $\tilde{H}_1 = \sum_{i=1}^{D-1} 1.2^i H_1$ where H_1 is the holding cost rate for the first stage of the supply chain network, per unit good. The average inventory in the make-to-stock portion is the steady state average number of tokens in places P_D and P_{rop}, call it N_D. Therefore the total holding cost is $H = \tilde{H}_1 \times N_D$.
2. *Lead time cost*: The time spent in assembling from the stock of semi-finished goods is taken for penalising late deliveries. Let H_2 be the average lead time cost rate in \$ per unit good per hour. The average throughout rate of the assemble to-order portion is the steady state average throughput rate of the transition t_N, say Λ_N. Let the steady state average number of orders in the assemble-to-order portion be K_n. (This is obtained as the average number of tokens in place P_N.) Therefore the total average lead time cost is obtained as $LT = H_2 \times 1/\lambda_N \times K_n$.

Hence the total cost is $TC = H + LT$. We can easily include back order costs; if desired. For ease of presentation, we ignore the same.

Let us consider a five stage supply chain, with the last stage being the retail outlet. The service rates at all the facilities are assumed to be generally distributed with given variances. Let the targeted finished goods at the decoupling point be 4. Let the number of Kanbans be one each in each stage of the assemble-to-order portion. This will mean that

Table 10 Input parameters for the DPLP considered

Facility	*Average service rate* (*jobs*/*h*) μ	*SCV of service time*
Supplier 1	10.0	0.7
Supplier 2	15.0	0.7
Initial assembly plant	25.0	0.8
Final assembly plant	10.0	0.8
Distribution	20.0	0.6

in the GSPN, the number of Kanbans circulating will be $N - D$ where N is 5. The input parameters for the DPLP considered are shown in Table 10. The logistics and interface rates are included within the facility service rates. Since we are not concerned with the routings within a facility, we assume that the service rates given as input parameters have been calculated by using known analytical or simulation models for each facility in isolation.

We consider the infinite demand case. As a consequence the transition t_k is immediate. As per our method, we proceed as follows. We set the value of D, the decoupling point, equal to 1, 2, 3, and 4. For each value, we solve the PFQNs in the make-to-stock and the assemble-to-order portion of the system and get the firing rates of transitions t_1 and t_N. (The values for the above rates are shown in Table 11.) Once this is done, we analyse the GSPN of Figure 7 by assuming that the above firing rates are exponential. Then the minimal cost solution is chosen as the decoupling point. The average inventory holding cost rates in the make-to-stock portion are varied in proportion with the average lead time cost rates in the assemble-to-order portion.

Trends for the total cost for various cost ratios, inventory control policies and back order cost effects are shown in Tables 12 and 13. The following are some of our observations:

- As the ratio H_2/H_1 increases, we see that the decoupling point is moving to the right. This conforms to a make-to-stock situation. Therefore when delayed customer order fulfilment is not desired, we stock finished goods at the distribution centre itself.
- As the ratio H_2/H_1 decreases, the decoupling point moves to the left, indicating that it is less costly to assemble-to-order rather than make the items to stock. This is due to the fact that it is costlier to stock finished goods as we move down the supply chain.

Our findings are in tune with those of Reference 17 in spirit, and are more general in the sense that we use general service times instead of normal distribution as assumed.

Table 11 Transition firing rates obtained from queueing analysis

Rate of t_1 (units/h)	*Decoupling point at*	
9.2357	1	
8.2948	2	
8.7374	3	
7.1111	4	
Rate of t_N (units/h)	*Decoupling point at*	*No. of tokens in place P_K*
3.7636	1	1
6.1293	1	2
7.4129	1	3
8.0841	1	4
4.7847	2	1
7.1048	2	2
8.0257	2	3
5.5679	3	1
7.6540	3	2
13.9392	4	3

Table 12 The total costs for general service times and base stock policy

Decoupling point	*Total cost*			
	$H_D/H_I = 10$	$H_D/H_I = 30$	$H_D/H_I = 40$	$H_D/H_I = 50$
1	3.596	5.940	8.285	10.630
2	5.214	6.963	8.712	10.461
3	8.967	10.237	11.508	12.779
4	12.071	12.429	12.788	13.147

Table 13 The total costs for general service times and reorder point policy

Decoupling point	*Total cost*			
	$H_D/H_I = 10$	$H_D/H_I = 20$	$H_D/H_I = 30$	$H_D/H_I = 40$
1	3.427	5.853	8.280	10.706
2	4.234	6.060	7.886	9.711
3	5.686	6.974	8.262	9.550
4	8.055	8.414	8.772	9.131

Conclusions

In this paper, we presented modelling techniques for analysing the supply chain process using generalised stochastic Petri nets (GSPN). Our models include the logistics subprocesses and also the interface subprocesses that exist between any two members of the supply chain. Using this framework we solved two important problems by formulating them as cost minimisation problems:

- Comparison of the make-to-stock and the assemble-to-order policies in a supply chain
- Locating the decoupling point in the supply chain.

Analytical modelling methodologies such as the one presented here provide insights complementary to those provided by simulation methods. The hierarchical modelling techniques used in Section 3 is particularly attractive because they are compact and are of reasonable fidelity. We assumed that all random variables are exponentially distributed. Other distributions can be considered but at the expense of computational complexity. The package used by us for solving Petri net models (SPNP) is very comprehensive. In cases where exponential models are grossly inadequate, the Petri net formalism can be used for guiding the simulation effort. Also a Petri net diagram provides insights into methods of controlling the material movement

and the workflow in a supply chain. It is possible to build Petri net based controllers similar to the cell controllers in automated manufacturing systems.[6]

References

1 Petri CA (1962). Kommunikation mit automaten. PhD thesis, Shriften des Institutes fur Instrumentelle Mathematik Bonn, Germany.
2 Lee HL and Sasser MM (1995). Product universality and design for supply chain management. *Prod Plan and Cont* **6**: 270–277.
3 Vollman TE, Berry WI and Whybark DC (1998). *Manufacturing planning and control systems.* The Dow Jones-Irwin/APICS Series in Production Management, Fourth Edition.
4 Erenguo SS, Simpson NC and Vakharia AJ (1999). Integrated production/distribution planning in supply chains: an invited review. *Eur J Opns Res* **115**: 219–236.
5 Silver AE, Pyke DF, and Peterson R (1998). *Inventory Management and Production Planning and Scheduling.* Wiley: New York.
6 Viswanadham N and Narahari Y (1992). *Performance Modeling of Automated Manufacturing Systems.* Prentice Hall: Englewood Cliffs, NJ.
7 Srinivasa Raghavan NR (1998). *Performance analysis and scheduling of manufacturing supply chain networks.* PhD Thesis, Indian Institute of Science, Bangalore.
8 Forrester JW (1961). *Industrial Dynamics.* MIT Press: Cambridge, MA.
9 Karmarkar US, Kekre S and Kekre S (1992). Multi-item batching heuristics for minimization of queueing delays. *Eur J Opl Res* **58:** 99–111.
10 Ciardo G, Muppala JK and Trivedi KS (1990). Manual for the SPNP Package Version 3.0. Technical report, Duke University, Durham, USA.
11 Puliafito A, Scarpa M and Trivedi KS (1998). Petri nets with *k* simultaneously enabled generally distributed timed transitions. *Perform Eval* **32**: 1–34.
12 Spearman MI and Zazanis MA (1992). Push and pull production systems: issues and comparisons. *Opns Res* **40**: 521–532.
13 Connors D, *et al* (1995). Dynamic modeling of re-engineering supply chains. Technical Report RC-19944, IBM Research Center, Almaden.
14 Cochran JK and Kim SS (1998). Optimum junction point location and inventory levels in serial hybrid push/pull production systems. *Int J Prod Res* **36**: 1141–1155.
15 Hodgson T and Wang D (1991). Optimal hybrid push/pull control strategies for a parallel multistage system. *Int J Prod Res* **29**: 1279–1287.
16 Whitt W (1983). The queueing network analyzer. *Bell Sys Tech J* **62**: 2779–2815.
17 Lee HL (1993). Design for supply chain management perspectives in operations management: Essays in honor of Elwood S. Buffa. Kluwer Academic Publishers: Boston, USA, pp 45–65.

Received February 1999;
accepted December 1999 after one revision