Total WIP and WIP Mix for a CONWIP Controlled Job Shop

Sarah M. Ryan*
Senior Member of IIE
Department of Industrial & Manufacturing Systems Engineering
Iowa State University
Ames, IA 50011-2164

F. Fred Choobineh
Fellow of IIE
Department of Industrial and Management Systems Engineering
University of Nebraska
Lincoln, NE 68588-0518

January, 2002

*Corresponding Author: smryan@iastate.edu; Voice: 515-294-4347; Fax: 515-294-3524

This is an Accepted Manuscript of an article published by Taylor & Francis in IIE Transactions in 2003, available online: http://dx.doi.org/10.1080/07408170304387

Total WIP and WIP Mix for a CONWIP Controlled Job Shop

Abstract

A planning procedure to set the constant level of work in process (WIP) for each product type in a job shop operated under CONWIP control is developed. We model the job shop as a single chain multiple class closed queuing network. Given a specified product mix and a total WIP, a nonlinear program bounds the throughput of the network and optimizes the WIP mix. We identify the minimum total WIP that is guaranteed to yield throughput near the maximum possible for the specified product mix and set individual WIP levels by multiplying the optimal WIP mix proportions by the minimum total WIP. Numerical examples illustrate how these individual product WIP levels achieve the goal of high throughput consistent with the specified product mix.

Importance of This Paper

The CONWIP control policy has been developed as an easily implementable alternative to pure pull policies for shop floor control. In contrast to pure pull policies, for which much research has concerned the appropriate number of production authorizations (kanbans) to place in circulation, CONWIP research on this decision has been relatively scant. In addition, the development and analysis of this control policy has focused on flow lines or assembly systems. Many of the benefits of CONWIP should accrue in job shops as well; however, in order to implement it, one must determine the total amount of WIP allowed in the system, as well as the quantity for each product (the WIP mix).

The designer of a pull policy for a job shop must also decide whether kanbans are shared among the products or whether individual WIP levels are fixed for each product. This issue is currently unresolved. We believe that the latter policy has more evidence to support it and that it is more consistent with the reasoning underlying the development of CONWIP control: that WIP levels are easy to control, while throughput (in particular, the production ratios for the different products) should be observed as an output variable.

However, the allocation of a fixed total WIP to different products is a difficult combinatorial problem. Our procedure to determine the total WIP and WIP mix uses a shared kanban model for planning purposes. For a given product mix the procedure uses nonlinear and linear programs to bound the system throughput and suggest the total WIP and the WIP mix proportions. Individual product WIP levels are extracted from this solution. Numerical examples illustrate the tightness and robustness of the throughput bounds and show how the resulting WIP levels achieve high throughput consistent with the desired product mix.

1 Introduction

By specifying the number of production authorizations (kanbans) in circulation on a manufacturing shop floor, pull control policies establish an upper bound for the work in process (WIP) since a fixed number of parts is associated with each kanban. Limiting the WIP in a manufacturing system potentially reduces jobs' flow times and storage space, and permits a rapid response to quality problems encountered on the shop floor. In addition, pull policies let demand govern the production system.

The pure pull policy specifies the number of kanbans between each pair of adjacent work stations. A more recent variation of pull control specifies the number of kanbans circulating among a larger set of work stations so that jobs are pulled into the set of stations and pushed among them. This variation, known as the CONWIP control policy (Spearman et al. 1990), is a combination of push and pull policies that, while easier to implement, provides most of the benefits of a pure pull policy.

Research on pull policies in general and CONWIP in particular has primarily focused on flow lines that produce one or more similar products (see, for example, Bard and Golany, 1991; Spearman and Zazanis, 1992; Spearman, 1992; Tayur, 1993; Gstettner and Kuhn, 1996; Dar-El et al., 1999). However, the benefits of a WIP limit can also be expected to accrue in job shops, in which multiple products with distinct processing requirements compete for the same set of resources. Consistent with the increased complexity of a job shop environment, implementing a pull policy in a job shop is significantly more complicated than in a flow shop. First, the pull system planner must determine a total WIP level to achieve high total throughput. This problem has received scant attention in the CONWIP literature since, in single product systems, the profitability of the system has been shown to be relatively insensitive to the WIP level (Spearman and Zazanis 1992).

Hence, most researchers have extended this characteristic to multiple product systems.

Second, the pull policy designer must decide whether kanbans are shared among the products or whether a number of kanbans is dedicated to each product. Shared kanbans can be modeled using a single chain closed queuing network as in Chevalier and Wein (1993). When a job has completed processing, a new job is released into the system, whose type is determined probabilistically according to the product mix. This release mechanism matches the individual throughput proportions to the product mix. In this approach, the WIP mix is not controlled directly. Instead, it varies over time in response to stochastic processing times and queuing interactions. We believe this approach runs counter to the reasoning underlying the development of CONWIP control: that WIP levels are easy to control, while throughput should be observed as an output variable.

When kanbans are dedicated to each product, individual product WIP levels are controlled and the system can be modeled as a multiple chain closed queuing network. When a job has completed processing, it is replaced by a new job of the same type as in Duenyas (1994). However, optimizing individual WIP levels directly is not straightforward. For example, Buzacott and Shanthikumar (1993, p. 392) have pointed out that when using separate WIP levels, the total throughput is not guaranteed to increase with the total WIP because the throughput of one product may drop when the WIP level for a different product is increased.

The question of single chain vs. multiple chain operation, i.e., shared vs. dedicated kanbans, has not been fully resolved and it is not the purpose of this paper to do so. Past analytical and simulation studies have measured performance by the WIP required to achieve individual or total throughput targets. Considered together, they suggest that a single WIP level with shared kanbans is better when products share similar routings, while individual WIP levels are preferred when different products have disparate routings. For example, evidence from simulations by Duenyas

(1994) favored multiple chain operation and he argued that its superiority was stronger when routings differed. On the other hand, simulation experiments of a multiple product serial system by Krishnamurthy et al. (2000) indicated that shared kanbans achieve a target *total* throughput level with less WIP than dedicated kanbans do when all products share the same routing. In addition, the example of the throughput's nonmonotonocity in Buzacott and Shanthikumar (1993) featured two products with very similar part routings. Since job shops typically have *dissimilar* part routings, in this paper we assume that the system will be operated with an individual WIP level for each product. This approach is consistent with Hopp and Roof (1998), who used shared kanbans when multiple products shared the same routing and dedicated kanbans when routings were different.

Assuming, then, that separate card counts are maintained for each product, the total WIP must be allocated to the different products. When producing multiple products, maximizing total throughput is not a sufficient objective for meeting competing product demands. Indeed, the highest total throughput might be achieved by producing only one of the products. To meet demand for all the products, one must proportionally match production rates to the required demand rates. The proportion of total WIP allocated to each product, the *WIP mix*, that optimizes the system performance generally will not be the same as the *product mix*. The product mix, specified by the proportions of the total demand on the system that correspond to each product, often is known, whereas the WIP mix should be influenced by the products' work contents and the capacities of processing resources. We assume that relative demand rates are stable within the planning horizon so that the product mix is a known parameter vector while the WIP mix is a vector of decision variables.

Finally, the performance of a control policy depends on the sequencing rules used at each station. For example, Chevalier and Wein developed a static preemptive priority sequence at each

station that can increase the total throughput when kanbans are shared among the products. The highest priority queue at each station is found by constructing a polytope whose vertices correspond to the pairs of products and stations, and then finding the subset of pairs, one for each station, whose vertices generate the simplex with highest volume-to-surface ratio. Under such a policy the WIP levels for products receiving lower priority in the sequencing at each station will be comparatively very high. In this paper we focus on easily implementable sequencing policies such as first come first served or random selection, recognizing that a more information intensive seqencing rule may improve system throughput.

In a notable extension of the CONWIP concept for a multiproduct factory, Suri (1998) proposed the POLCA control system. POLCA, which stands for Paired-cell Overlapping Loops of Cards with Authorization, assumes that the factory has been partitioned into non-overlapping manufacturing cells. POLCA maintains constant WIP between every pair of cells that experience inter-cell part movement over a planning horizon. Part release to a cell requires an appropriate kanban card as well as an authorization from the factory loading system. Zhou et al. (2000) used simulation to show that both CONWIP and POLCA achieve a better tradeoff of total WIP and total throughput than either pure push or pure pull policies.

In other multiproduct CONWIP related work, Golany et al. (1999) studied CONWIP control for a multiproduct system in which the products have been grouped into families and the machine groups into cells. They formulated and heuristically solved a mathematical program to determine the optimal allocation of a fixed number of kanbans (containers) to cells. In numerical examples, lower flow times were obtained if containers migrated freely among cells. Choobineh and Sowrirajan (1996) developed a hybrid approach. The total WIP was specified but the WIP level for each product could vary dynamically within heuristic limits specified according to the prod-

uct mix and the work content of each product. By simulation they showed that total throughput would benefit from reserving some capacity for each product. Hopp and Roof (1998) developed a procedure to dynamically adapt card counts to meet target throughput rates based on empirical observations of throughput. Luh et al. (2000), working with an aircraft company's job shop, formulated and approximately solved an optimization model to schedule a set of jobs over a specified time horizon to meet fixed due dates for a given total shared WIP level, assuming deterministic processing times. Ryan et al. (2000) developed a heuristic allocation procedure to determine the total WIP and WIP mix to satisfy a uniform service level across product types under the multiple chain control policy. Assuming heavy demand, they modeled the job shop as a closed queuing network and used an approximate performance evaluation to determine a WIP total and mix that would achieve a high throughput consistent with the product mix. When tested with randomly arriving demands following the product mix, the WIP mix did provide balanced customer service, as measured in the approximate performance evaluation by the proportion of arriving orders that wait to be fulfilled. Numerical examples showed that the most effective WIP mix is often quite different from the specified product mix.

In this paper we propose a systematic procedure for identifying the total WIP and WIP mix for a job shop operating under the CONWIP control policy. Previous methods for setting individual product card counts have used a heuristic allocation procedure that called upon simulation, an approximate performance evaluation, or empirical observation to estimate the throughput (Duenyas, Ryan et al., Hopp and Roof). The approximate performance evaluation methods lack any guarantee of accuracy, and the design of an efficient and accurate allocation heuristic is hampered by nonseparability of the individual product throughputs and nonmonotonicity of the total throughput. Target throughput levels that were set *a priori* occasionally must be adjusted for feasibility. Simi-

larly to Ryan et al. (2000), we use a closed queuing network to model the job shop; however, here we drop the assumption of heavy demand. Rather than approximating the throughput, we develop mathematical programs whose optimal objective values bound the throughput of the single-chain network from above and below. The bounding method can handle complex routings such as repeat visits to a station with different processing rates. Since, in practice, the operational capacity of a job shop is set to be some fraction of its nominal capacity (typically 85% to 95%), the desirable total number of jobs for the network is identified when the lower bound on the throughput is a proportion $\beta < 1$ of the upper bound. For the desired WIP mix we use the individual product average WIP proportions that maximize the single-chain throughput for the desired total WIP. Numerical tests with simulation in a multiple-chain model show that these derived WIP levels achieve high throughput consistent with the specified product mix.

In the next section we introduce notation and outline the solution approach. In Section 3 we describe in more detail the queuing network model of the job shop operating under single chain CONWIP control and develop the mathematical programs to bound the throughput and optimize the WIP mix. In Section 4, the bounds are explored and validated by simulation using a small system, and the impact of sequencing rules is discussed. Our approach for setting the WIP level and applying the WIP mix is illustrated on a larger system. We conclude with Section 5.

2 Outline of Approach

Nomenclature:

$P =$ the number of distinct products to be manufactured.

$\alpha =$ the specified product mix vector, where $\sum_{r=1}^{P} \alpha_r = 1$. If orders for product r arrive randomly at mean rate λ_r, then the product mix vector can be derived by setting $\alpha_r = \lambda_r / \sum_{j=1}^{P} \lambda_j$.

$\beta =$ a specified proportion of the maximum throughput.

$N =$ the total number of kanbans.

$\theta_r^S(N) =$ the mean throughput of type r products when the system is operated as single chain (with shared kanbans), $r = 1, \ldots, P$.

$\Theta^S(N) =$ the mean total throughput when operated in single chain mode; $\Theta^S(N) = \sum_{r=1}^{P} \theta_r^S(N)$.

$\Theta_{LB}^S(N), \Theta_{UB}^S(N) =$ lower and upper bounds, respectively, on the single chain throughput.

$Y_r(N) =$ the average number of type r jobs in the system that maximizes $\Theta^S(N)$, $r = 1, \ldots, P$.

$p_r(N) =$ the proportion of type r jobs in the single chain optimal WIP mix; $p_r(N) = Y_r(N)/N$.

$K_r =$ number of kanbans for type r products when the system is operated in multiple chain mode, $r = 1, \ldots, P$.

$\theta_r^M(K_1, \ldots, K_P) =$ the mean throughput of type r products in multiple chain operation.

$\Theta^M(K_1, \ldots, K_P) =$ the mean total throughput in multiple chain operation, $\Theta^M(K_1, \ldots, K_P) = \sum_{r=1}^{P} \theta_r^M(K_1, \ldots, K_P)$.

$\gamma_r(K_1, \ldots, K_P) = \theta_r^M(K_1, \ldots, K_P)/\Theta^M(K_1, \ldots, K_P) =$ the rth element of the multiple chain throughput mix vector.

As stated above, we assume the system will be operated with a fixed number of kanbans dedicated to each product type. Our goal is to determine these fixed numbers. However, the lack of monotonicity in the throughput as well as the combinatorial complexity of allocating WIP to products discourage the direct optimization of kanban allocation. Instead, we use a two stage process. In stage one, we use a single chain queuing network model for shared kanbans to discover a total WIP and WIP mix that will achieve high total throughput consistent with the required product mix. In stage two we use the results of stage one to determine individual product WIP levels. In a single chain network, the total throughput does increase monotonically with the total amount

of WIP (Suri 1985). The total WIP is held constant while the type of job released to the system is governed probabilistically according to the product mix. For maximum applicability, we also allow deterministic or probabilistic routing with different exponential processing time distributions for different classes at the same station. Repeat visits to the same station may occur either for quality related rework or by ordinary routing specification, with possibly a different processing rate on each visit. Even when processing times are limited to exponential distributions, this single chain multiple class queuing network will not have a product form solution. Our performance evaluation method is adapted from Kumar and Kumar (1994) and enhanced to yield tighter bounds assuming first come first served or random rather than preemptive priority sequencing. The performance evaluator, a nonlinear program (NLP), provides upper and lower bounds on the throughput to result from a given product mix and WIP level. From the convergence of the bounds, we can identify the smallest WIP level that guarantees some specified percentage of the maximum achievable single chain throughput for a given product mix. We extract the optimal WIP mix from the upper bounding solution of the nonlinear program. Figure 1 depicts the two stages of our approach.

***** Figure 1 Here *****

Specifics of the two stage process depicted in Fig. 1 are as follows:

1. Model the job shop in single chain mode and develop its NLP performance model. Vary N and solve the NLP to bound the throughput of the system. Since the type of each job released to the system is selected probabilistically according to α, $\theta_r^S(N) = \alpha_r \Theta^S(N)$. Find $N' = \min\{N : \Theta_{LB}^S(N) \geq \beta \Theta_{UB}^S(N)\}$. Extract $Y_r(N')$ from the NLP solution that achieves $\Theta_{UB}^S(N')$ and compute $p_r(N')$.

2. Round the values of $p_r(N')N'$ up or down as needed to obtain integer values of K_r such that $\sum_{r=1}^{P} K_r = N'$. In the resulting multiple chain operation, we can verify by simulation that

$\gamma_{\mathsf{r}}(K_{\mathsf{1}}, \ldots, K_{\mathsf{P}}) \cong \alpha_{\mathsf{r}}$ for each r while, typically, $\Theta^{\mathsf{M}}(K_{\mathsf{1}}, \ldots, K_{\mathsf{P}}) > \Theta^{\mathsf{S}}(N')$.

Since the value of N is determined by distance between the bounds, it could be larger than necessary to achieve the desired throughput in operation. However, the optimal WIP mix in numerical examples is fairly stable as N increases. In practice, the values of $K_1, \ldots, K_{\mathsf{P}}$ obtained by our procedure could be used as a starting point for experimentation in the actual or simulated system to see if card counts could be reduced without significantly lowering the individual throughputs or altering the balance among them. The desired product mix α, the WIP mix p derived from the model, and the resulting throughput mix γ are all nonnegative P-vectors whose elements sum to 1. We normalize Euclidean distances between any pair by dividing by the maximum possible distance between them (equal to $\sqrt{2}$). We use this metric in Section 4.2 to compare WIP mixes found with different values of N and to examine the match between γ and α for alternative allocations of the same N.

3 Queuing Network Model and Performance Evaluation

The following describes additional notation used in the queuing network model of the job shop.

$S =$ the number of single server processing stations in the job shop.

$I =$ the index of a station that is visited by all products, $1 \leq I \leq S$.

$R =$ the number of customer classes in the queuing network model of the job shop, $R \geq P$. We may use a probabilistic routing to model the average effect of utilizing alternative routings. In addition, a job may visit the same station more than once, either in the normal course of processing or for quality related rework. The first P classes correspond to the product types, and the additional $R - P$ classes are artifices to model repeat visits to the same station(s).

$q_{\mathsf{rsjv}} =$ the probability that a class r customer, having completed processing at station s, joins the

queue at station v as a class j customer. Assume $q_{\mathsf{rsrs}} = 0$. If product r has a single process plan (route), then $q_{\mathsf{rsrv}} = 1$, where v is the next station after s in product r's routing. If product r revisits station s, we create a new customer class $j > P$ to model the reentrants and set $q_{\mathsf{rsjv}} = 1$, where v is the station visited by product r immediately after the first visit to station s.

F_{r} (resp. L_{r}) = the first (last) station visited by customer class r.

μ_{rs} = mean processing rates for class r customers at station s. Processing times are exponentially distributed.

For a given value of N, the job shop under single chain CONWIP control is modeled as a closed queuing network in which the customers are the N kanbans, each of which adopts the class of the product to which it is currently attached. (In this section, we suppress dependence on N as well as the single chain superscript S from the notation). Upon completion of processing, a product's kanban is surrendered and immediately attached to a new job of type j with probability α_{j}. Thus for each r, $q_{\mathsf{rL_rjF_j}} = \alpha_{\mathsf{j}}$, for $j = 1, \ldots, P$. This release policy ensures that $\theta_{\mathsf{r}} = \alpha_{\mathsf{r}}\Theta$ for $r = 1, \ldots, P$. We assume an infinite supply of raw material for each product type, and unlimited waiting space both within the shop and for finished products.

The optimal number of jobs in a closed manufacturing network has been addressed in several papers (Vinod and Solberg, 1985; Dallery and Frein, 1988; Lee et al., 1989; Kouvelis and Kiran, 1991; Kouvelis and Lee, 1995) in combination with other aspects of designing flexible manufacturing systems. These works considered a single product type. In order to implement multiproduct CONWIP, we wish to determine a small value of N that will yield a high throughput. However, for a given N, an exact evaluation of the network's performance characteristics is not possible. Because of the distinct processing time distributions for different customer classes at a station without

processor sharing, this closed queuing network does not have a product form solution. The single chain control policy precludes the use of approximate multiclass mean value analysis (Schweitzer et al. 1986) as well as the decomposition approximation of Baynat and Dallery (1996). The throughput bounds of Kerola (1986) also require a fixed population for each class. The generating function technique of Reiser and Kobayashi (1980) allows the single chain policy, but restricts the queue discipline to processor sharing or preemptive resume. The performance evaluator used by Ryan et al. (2000) does not allow different processing times for repeat visits. In this paper, we use an adaptation of the performance evaluation method of Kumar and Kumar (1994), which was also developed independently by Bertsimas et al. (1994).

We present nonlinear and linear programs (NLP and LP) to bound the system throughput. For fixed N, $\Theta^{S}_{LB}(N)$ (resp. $\Theta^{S}_{UB}(N)$) is obtained by minimizing (maximizing) system throughput, subject to constraints on (1) the constant system population, (2) "sampling equalities" (Jin et al. 1997), (3) non-idling, (4) stationary first moments of queue lengths, (5) stationary second moments of queue lengths, (6) sojourn times at each station, and nonnegativity restrictions. All elements of this model except (6) are adapted from Kumar and Kumar (1994) and Jin et al. (1997). Lower and upper bounds on the throughput for a given N are obtained by minimizing and maximizing the objective function, respectively. When using only constraints (1) - (5), a large gap between the lower and upper bounds is often observed. Kumar and Kumar (1994) and Bertsimas et al. (1994) incorporated additional constraints for preemptive priority policies to reduce the gap. However, preemptive priority is rare in manufacturing practice. In this paper, we take a more practical manufacturing approach to tighten the bounds. We develop constraints (6) to balance the mean waiting time at each station under first come first served sequencing as is common in models of CONWIP systems (see, for example, Herer and Masin (1997) for a different mathematical

programming formulation of a serial CONWIP system). Because the nonlinear constraints (6) can require long solution times, we also develop approximate linear versions.

The variables for the mathematical programs are defined in terms of expected values of two families of stochastic processes:

$X_{rs}(t) =$ the number of class r customers at station s (in queue or in service) at time t, and

$W_{rs}(t) = 1$ if the station s server is busy with a class r customer at time t and 0 otherwise. Let

$\rho_{rs} = E[W_{rs}(t)]$, and

$z_{rsjv} = E[W_{rs}(t) X_{jv}(t)]$ for any t (in steady state).

These processes are defined only for pairs (r, s) and (j, v) such that class r visits station s and class j visits station v. Let L be the number of such class-station pairs. We also define $R(s)$ as the set of customer classes that visit station s. In the following, the sum over classes $\sum_j$ will signify $\sum_{j \in R(s)}$ where not ambiguous.

The decision variables in the LP/NLP are ρ_{rs} and z_{rsjv}. The variables ρ_{rs} can be interpreted directly as the proportion of time station s spends serving class r customers. The interpretation of the cross moment variables z_{rsjv} is less straightforward. These variables are required to express the constraints (5) below on the stationarity of the second moments of the queue lengths, $X_{rs}(t)$.

3.1 A Linear Program to Bound Throughput

The throughput of class r at station s may be expressed as $\mu_{rs}\rho_{rs}$. If we select a station $s = I$ that is visited by all customer classes, such as a loading/unloading station for a flexible manufacturing system, the objective function is given by:

$$\Theta = \sum_{r=1}^{R} \mu_{rI}\rho_{rI}$$

We assume a non-idling policy; that is, if any customers are present at station s, the station s server must not be idle. Mathematically, this assumption is expressed as $X_{rs}(t) > 0 \Rightarrow W_{js}(t) =$

1 for some $j \in R(s)$. Then $X_{rs}(t) = \sum_j W_{js}(t) X_{rs}(t)$ and we can write $E[X_{rs}(t)] = \sum_j E[W_{js}(t) X_{rs}(t)] = \sum_j z_{jsrs}$. Also note that the total population of type r jobs is given by $Y_r = \sum_{s=1}^{S} \sum_t \sum_j z_{jsts}$, where the sum over t is over all classes $t \in \{1, \ldots, R\}$ that pertain to job type r, including reentrants.

The first constraint stipulates that $\sum_{r,s} E[X_{rs}(t)] = N$ for all t, or

$$\sum_{s=1}^{S} \sum_{r=1}^{R} \sum_j z_{jsrs} = N. \tag{1}$$

This constraint actually limits the average WIP, not the maximum WIP. However, since $\sum_{(j,v)} X_{jv}(t) = N$ (not only in expected value), it follows that $E[W_{rs}(t) \sum_{(j,v)} X_{jv}(t)] = \rho_{rs} N$, so that the following stronger constraints also hold:

$$\sum_{(j,v)} z_{rsjv} - N\rho_{rs} = 0, \forall (r,s). \tag{2}$$

The non-idling assumption leads to another set of constraints on the z variables as in Kumar and Kumar (1994). Since $\sum_j W_{jv}(t) \leq 1, \forall v$, we have $X_{rs}(t) = \sum_j W_{js}(t) X_{rs}(t) \geq \sum_j W_{jv}(t) X_{rs}(t)$. Taking expectations on both sides of the inequality yields:

$$\sum_j z_{jvrs} - \sum_j z_{jsrs} \leq 0, \forall (r,s), v \neq s. \tag{3}$$

Constraints on station utilization could be given as $\sum_r \rho_{rs} \leq 1, \forall s$; however, these are implied by constraints (1) - (3).

The ordinary traffic equations for the closed queuing network are a result of the assumption that the first moments of the populations at each station are stationary, i.e., that $E[X_{rs}(t)]$ is constant for all t in steady state. These may be expressed as:

$$-\mu_{rs}\rho_{rs} + \sum_{(j,v)} q_{jvrs}\mu_{jv}\rho_{jv} = 0, \forall (r,s). \tag{4}$$

Let $\delta_{rsjv} = 1$ if $(r,s) = (j,v)$, and 0 otherwise. Assume that the buffers are lexicographically ordered, so that $(j,v) > (r,s)$ if either $j > r$ or $j = r$ and $v > s$. Assuming that the second mo-

ments, $E[X_{rs}^2(t)]$, and cross moments, $E[X_{rs}(t)\,X_{jv}(t)]$, of the class populations at each station are stationary, and substituting equation (4) yields:

$$\begin{aligned} \sum_{(k,y)} q_{kyrs}\mu_{ky}z_{kyjv} + \sum_{(k,y)} q_{kyjv}\mu_{ky}z_{kyrs} - \mu_{rs}z_{rsjv} - \mu_{jv}z_{jvrs} & \\ -q_{rsjv}\mu_{rs}\rho_{rs} - q_{jvrs}\mu_{jv}\rho_{jv} + 2\delta_{rsjv}\mu_{rs}\rho_{rs} &= 0, \qquad (5) \\ \forall(r,s), \forall(j,v) &\geq (r,s). \end{aligned}$$

Details of the derivation can be found in Kumar and Kumar (1994) or Jin et al. (1997).

3.2 Nonlinear Constraints to Tighten the Bounds

Constraints (1) - (5) concern the global operation of the system and the flow of customers between stations. The final set of constraints deals with the competition among customer classes at each station. Schweitzer et al. (1986) treat a closed multiple chain queuing network with a fixed population K_r of class r customers. Assuming first come first served sequencing, they show that for each (r,s), the expected sojourn time spent by class r customers at station s is given by

$$D_{rs} = \frac{1}{\mu_{rs}} + \sum_{j} \frac{1}{\mu_{js}} E[X_{js}] + \varepsilon_{rs},$$

where ε_{rs} is a correction term that arises from the disparity between the arrival-average and time-average queue lengths. This equation simply states that an arriving customer's sojourn time is the sum of its own service time and the service times of customers ahead of it in the queue. Reiser and Lavenberg (1980) showed that the relation holds for last come first served preemptive sequencing as well. The correction term $\varepsilon_{rs} = -E[X_{rs}]/(\mu_{rs}K_r)$ is designed to avoid "double-counting" the service time of an arriving customer in its sojourn time. Applying Little's Law, we can write $D_{rs} = E[X_{rs}]/(\mu_{rs}\rho_{rs})$, so that in terms of our decision variables we have

$$\sum_{j} z_{jsrs} = \rho_{rs} + \mu_{rs}\rho_{rs} \sum_{j} \frac{1}{\mu_{js}} \sum_{k} z_{ksjs} + \mu_{rs}\rho_{rs}\varepsilon_{rs}.$$

Since $\varepsilon_{rs} < 0$, and clearly $\varepsilon_{rs} > -1/\mu_{rs}$, we arrive at two sets of *nonlinear* constraints:

$$\sum_j z_{jsrs} \leq \rho_{rs} + \mu_{rs}\rho_{rs} \sum_j \frac{1}{\mu_{js}} \sum_k z_{ksjs}, \forall (r,s) \tag{6NA}$$

$$\sum_j z_{jsrs} \geq \mu_{rs}\rho_{rs} \sum_j \frac{1}{\mu_{js}} \sum_k z_{ksjs}, \forall (r,s) \tag{6NB}$$

3.3 Linear Versions of Bound-Tightening Constraints

We can also derive approximate linear constraints. Recall that $\alpha = (\alpha_1, \ldots, \alpha_P)$ with $\sum_{r=1}^{P} \alpha_r = 1$ is a vector that describes the desired apportionment of throughput to the products. The single chain release policy, embodied in the routing probabilities $q_{rL_rjF_j} = \alpha_j$, guarantees that the overall product throughputs proportionally follow the vector α. Now we make the stronger assumption that the same throughput proportions hold at each station. For a class $r > P$ that represents reentrants of type k products, we set $\alpha_r = \alpha_k$. At station s, let $\alpha_{rs} = \alpha_r / \sum_{j \in R(s)} \alpha_j$ be this proportion adjusted for the set of classes $R(s)$ that visit station s. Assume that

$$\mu_{rs}\rho_{rs} = \alpha_{rs} \sum_{j \in R(s)} \mu_{js}\rho_{js}, \forall (r,s).$$

Also, let

$$M_{ks} = \max_{j \in R(s)} \left\{ \frac{\mu_{js}}{\mu_{ks}}, \right\} \forall (k,s).$$

Then, on the right hand side of inequality (6NA), we have

$$\begin{aligned}
\mu_{rs}\rho_{rs} \sum_j \frac{1}{\mu_{js}} \sum_k z_{ksjs} &= \alpha_{rs} \left(\sum_j \mu_{js}\rho_{js} \right) \left(\sum_j \frac{1}{\mu_{js}} \sum_k z_{ksjs} \right) \\
&= \alpha_{rs} \sum_j \sum_k \rho_{js} \frac{\mu_{js}}{\mu_{ks}} E[X_{ks}] \\
&\leq \alpha_{rs} \sum_j \rho_{js} \sum_k M_{ks} E[X_{ks}] \\
&\leq \alpha_{rs} \sum_k M_{ks} \sum_j z_{jsks}
\end{aligned}$$

Thus we arrive at the set of linear constraints:

$$\sum_{j} z_{jsrs} \leq \rho_{rs} + \alpha_{rs} \sum_{k} M_{ks} \sum_{j} z_{jsks}, \forall (r,s) \tag{6LA}$$

We can also derive a linear version of inequalities (6NB) for bottleneck stations in saturated systems. Let

$$m_{rs} = \min_{j \in R(s)} \left[\frac{\mu_{rs}}{\mu_{js}}, 1 \right], \forall (r,s).$$

Then, on the right hand side of (6NB),

$$\begin{aligned} \mu_{rs}\rho_{rs} \sum_{j} \frac{1}{\mu_{js}} \sum_{k} z_{ksjs} &\geq \rho_{rs} m_{rs} \sum_{j} \sum_{k} z_{ksjs} \\ &= m_{rs}\rho_{rs} \left(\sum_{j} z_{rsjs} + \sum_{k \neq r} \sum_{j} z_{ksjs} \right). \end{aligned}$$

Assuming that sequencing at a bottleneck station is independent of the total number of jobs waiting, we have

$$\frac{\sum_{k \neq r} \rho_{ks}}{\rho_{rs}} = \frac{\sum_{k \neq r} E[W_{ks}(t)] E\left[\sum_{j} X_{js}(t)\right]}{E[W_{rs}(t)] E\left[\sum_{j} X_{js}(t)\right]} = \frac{\sum_{k \neq r} \sum_{j} z_{ksjs}}{\sum_{j} z_{rsjs}}.$$

Substituting into (6NB), we obtain

$$\begin{aligned} \sum_{j} z_{jsrs} &\geq m_{rs} \left(\rho_{rs} \sum_{j} z_{rsjs} + \sum_{k \neq r} \rho_{ks} \sum_{j} z_{rsjs} \right) \\ &= m_{rs} \sum_{j} z_{rsjs}, \text{ if } \sum_{k} \rho_{ks} = 1. \end{aligned}$$

The final set of linear constraints, then, which are valid only for bottleneck stations in saturated systems, are:

$$\sum_{j} z_{jsrs} \geq m_{rs} \sum_{j} z_{rsjs}, \forall (r,s) \text{ with } \sum_{k} \rho_{ks} = 1. \tag{6LB}$$

3.4 Summary of Mathematical Programs

This planning model differs in one important aspect from previous work. Rather than using priority sequencing at each station, we assume that sequencing at each station is independent of product type. The throughput bounds nevertheless encapsulate simulation results obtained with a variety

of sequencing rules, including static and dynamic priority rules. A preemptive priority policy can provide tight throughput bounds as in Kumar and Kumar (1994), and can yield good *total* throughput results as in Chevalier and Wein (1993). However, our simulation results indicate that the impact of the scheduling policy on the total throughput diminishes as the total WIP increases. Moreover, priority sequencing severely skews the WIP mix toward lower priority product types. Nonpreemptive FCFS sequencing is more likely to be implemented in actual manufacturing settings and more consistent with our goal of *balancing* the throughput across different product types.

Thus we have:

1. a nonlinear model consisting of the linear objective (0), linear constraints (1)-(5) and nonlinear constraints (6NA&B),

2. a linear model that includes the objective (0) and linear constraints (1)-(5), (6LA&B).

There are a total of $L^2 + L$ decision variables. The nonlinear model includes $L^2 + 3L + S + 1$ constraints, while the linear model includes $L^2 + 2L + S + BR + 1$ constraints, where B is the number of bottleneck stations (typically one).

3.5 Finding Total WIP and WIP Mix

The goals of developing both the nonlinear and linear models are to help choose a good value of N, the total WIP, and to predict the WIP mix. Based on the monotonicity of the single chain throughput in N, our approach is to choose some proportion of the upper bound on throughput, say $\beta = 0.95$, and find the smallest N, call it N', for which a lower bound for the throughput exceeds this quantity. The nonlinear model provides increasingly tight bounds on the throughput, $\Theta_{\mathrm{LB}}^{\mathrm{S;NLP}}(N)$ and $\Theta_{\mathrm{UB}}^{\mathrm{S;NLP}}(N)$, as N increases but it may require long computation times for large job shops. Also, since the nonlinear constraints are not convex, it is not guaranteed to yield global op-

tima. The linear model is solved very quickly to optimality but its bounds, $\Theta_{\mathrm{LB}}^{\mathsf{S};\mathrm{LP}}(N)$ and $\Theta_{\mathrm{UB}}^{\mathsf{S};\mathrm{LP}}(N)$, are not as tight. However, the distance $\Theta_{\mathrm{UB}}^{\mathsf{S};\mathrm{LP}}(N) - \Theta_{\mathrm{LB}}^{\mathsf{S};\mathrm{LP}}(N)$ stabilizes with increasing N. This stability can be observed as an approximate indicator that the marginal improvement in throughput is diminishing.

We propose the following heuristic procedure to find N' without excessive computation time. This heuristic uses the solutions of the NLP as a starting solution. It uses the LP to explore the neighborhood of the starting solution in order to improve it. Finally, the NLP is used again to refine the solution. The steps of the heuristic are as follows:

1. Maximize and minimize the *nonlinear* model for some N_0 both to obtain initial bounds for the throughput and to determine which station is the bottleneck. Let $\beta_0 = \Theta_{\mathrm{LB}}^{\mathsf{S};\mathrm{NLP}}(N_0)/\Theta_{\mathrm{UB}}^{\mathsf{S};\mathrm{NLP}}(N_0)$.
2. Maximize and minimize the *linear* model, including constraint (6LB) for the bottleneck, for several values of N above (below) N_0 if β_0 is below (above) β. Let N'' be the smallest N for which the distance $\Theta_{\mathrm{UB}}^{\mathsf{S};\mathrm{LP}} - \Theta_{\mathrm{LB}}^{\mathsf{S};\mathrm{LP}}$ for $N+1$ is less than, say, 0.5 percent smaller than that for N.
3. Use the *nonlinear* model to explore values near N'' until the smallest N' is found such that $\Theta_{\mathrm{LB}}^{\mathsf{S};\mathrm{NLP}}/\Theta_{\mathrm{UB}}^{\mathsf{S};\mathrm{NLP}}(N') \geq \beta$.

Our assumed order release mechanism guarantees that the desired product mix will be obtained in single chain mode. The WIP mix may be observed in the maximizing solution for either the LP or the NLP as $(p_1(N), \ldots, p_{\mathsf{R}}(N))$, where $p_{\mathsf{r}}(N) \equiv Y_{\mathsf{r}}(N)/N$ is the expected proportion of jobs in the system that are of type r. In computational experience, the nonlinear model has provided a more stable and accurate (compared with simulation) prediction of the WIP mix for various values of N.

Product	Routing	Proc. Times
1	1, 2, 4	1, 1, 1
2	1, 3, 4	$1/\mu, 1/\mu, 1/\mu$

Table 1. Routings and expected processing times for Example 1.

4 Numerical Examples

We use a small example system to explore and validate the model in stage 1 and a larger system to illustrate its use in stage 2. Example 1, with four stations and two products, was designed to systematically test the mathematical programs over a variety of product mixes and processing time relationships. Example 2, with ten stations, five products, and complex routings, demonstrates how the output of stage 1 may be used in practice for setting the individual product WIP levels in a multiple chain release policy. For each example, the bounds obtained by the linear and nonlinear models were compared to results obtained from simulation. The simulations, coded in SIMAN, were initialized with a transient period after which the mean throughput was calculated from 30 consecutive batches of 1,000 minutes each. From moving average plots of the queue lengths at the bottleneck station, a 300,000 minute transient period was judged to be sufficient for each example. The simulation was performed starting from an initial empty and idle condition, with N jobs waiting at station 1. Each entering job was assigned to class r with probability α_r, $r = 1, \ldots, P$. In the single chain simulation, when a job had completed processing, it was replaced by a new job at station 1, of class r with probability α_r.

4.1 Example 1.

The first example was studied in depth to answer: (1) How tight are the throughput bounds? (2) How reliable is the WIP mix? and (3) What is the impact of sequencing rules on the throughput and the WIP mix? This small example system has two products that share stations 1 and 4, while intermediate processing is done by station 2 for product 1 and station 3 for product 3. The routings

α_1	μ	$\Theta_{LB}^{S;LP}$	$\Theta_{LB}^{S;NLP}$	**CYC**	**FCFS**	**LNQ**	**RAN**	**SPT**	$\Theta_{UB}^{S;LP} = \Theta_{UB}^{S;NLP}$	$\Theta_{LB}^{S;NLP}/\Theta_{UB}^{S;NLP}$	$p_1(30)$
0.3	1	0.714	0.933	0.966	0.961	0.964	0.966	0.962	0.968	0.96	0.30
0.3	2	1.07	1.43	1.47	1.48	1.49	1.47	1.50	1.50	0.95	0.31
0.3	3	1.20	1.72	1.75	1.79	1.81	1.76	1.83	1.83	0.94	0.30
0.5	1	0.720	0.934	0.967	0.963	0.965	0.964	0.965	0.968	0.96	0.50
0.5	2	0.891	1.24	1.28	1.28	1.28	1.28	1.29	1.30	0.95	0.53
0.5	3	0.938	1.38	1.43	1.44	1.43	1.43	1.46	1.46	0.95	0.54
0.7	1	0.714	0.933	0.966	0.965	0.964	0.969	0.960	0.968	0.96	0.72
0.7	2	0.770	1.09	1.14	1.13	1.12	1.13	1.14	1.14	0.96	0.71
0.7	3	0.782	1.14	1.20	1.19	1.18	1.20	1.20	1.21	0.94	0.74

Table 2. Bounds on total throughput obtained from the LP and NLP compared with simulation for N=30.

and mean processing times are shown in Table 1. The mean processing rate for product 1 at each station is 1 time unit per job, while the mean processing rate for product 2 at each station is a variable parameter, μ. Nine test cases are generated by having α_1 take on the values of 0.3, 0.5 or 0.7 (with $\alpha_2 = 1 - \alpha_1$) and μ vary among 1, 2 and 3. For this example, solving either the nonlinear or the linear model takes a few hundredths of a second using LINGO on a 200 MHz Pentium processor.

4.1.1 Throughput Bounds

For each of the nine combinations of values of α_1 and μ, the throughput bounds were compared to results obtained from simulation. Though constraints (6) were based on a first come first served assumption, we tested one dynamic and four static nonpreemptive sequencing rules:

- CYC: A fixed order is established for the queues at a station. Upon completing service on a customer from a given queue, the server examines each queue starting from the next queue in the fixed order and selects the first customer from the first nonempty queue encountered.
- FCFS: Customers are served in order of arrival, irrespective of their class.
- LNQ: Upon completing a service, the server begins service on the first customer from the longest queue.

- RAN: Upon completing a customer, the server randomly chooses the queue to serve next.
- SPT: Upon completing a service, the server always chooses the first customer from the queue with smallest expected processing time, if one is available.

For all nine combinations of parameter values in Example 1, the sequencing policy of Chevalier and Wein (1993) is a preemptive version of our SPT rule. Therefore, it is not explicitly identified in our experiments.

In each of the nine cases, with N varying from 10 to 40, the linear and nonlinear models both bound the simulation results for all the sequencing rules (see Table 2 for N=30). Each of the 30 simulation batches included approximately 30 completed jobs, following a warmup period of approximately 9000 completions. Fig. 2 shows the bounds obtained by the model and the simulation means for different sequencing rules and values of N in a typical case. The bounds, as well as the simulation results, appear increasing and concave in N as suggested by Suri (1985) and Shanthikumar and Yao (1988). The maximum throughputs obtained by the linear and nonlinear models ("Max") are identical, but the nonlinear model provides a much tighter lower bound ("NLP Min") than the linear program ("LP Min"). Though the station sojourn constraints, (6NA) and (6NB), or (6LA) and (6LB), are based on a particular sequencing assumption, varying the sequencing seems to account for much of the distance between the upper and lower bounds provided by the nonlinear model. The decrease of this distance with N and encapsulation of different sequencing rules indicates that the impact of the sequencing policy on the throughput diminishes as the system population increases. From these results we conclude that the nonlinear constraints are necessary to provide tight throughput bounds. Further, though we have no guarantee that the solver will find the global optima, the smooth progression of values as N increases, together with the consistent encapsulation of simulation results (for all the parameter values tested), give us much confidence

in the validity of the bounds.

***** Figure 2 Here *****

4.1.2 WIP Mix

The inventory mix in the system is far more sensitive to sequencing rules than is the throughput. We can use the models to predict the mix of work in process by examining the proportion of jobs of each type stipulated in the optimal solutions. That is, the expected proportion of jobs in the system that are of type r is given by $p_r(N) = \sum_s \sum_j z_{jsrs}/N$. Fig. 3 shows a plot of the proportion of type 1 jobs, $p_1(N)$, predicted by maximizing the linear and nonlinear models as well as the averages observed during simulation using different sequencing rules. The nonlinear model is consistent with the FCFS simulation and intermediate among the other sequencing rules, while the prediction from the linear model varies wildly as N increases.

***** Figure 3 Here *****

A parametric analysis of the linear model helps explain the inconsistent inventory mixes predicted by the linear model. A pair of constraints, $\sum_s \sum_j z_{jsrs} = Np_r, r = 1, 2$, was added to the model to specify the WIP mix. A parametric analysis was performed on the right hand sides with $p_1 + p_2 = 1$ while keeping N fixed. On the vertical axis of Fig. 4 the proportions of type 1 jobs in the system (p_1) observed in simulation for $\alpha = (0.7, 0.3)$, $\mu = 2$, and $N = 30$ are displayed and labeled with the sequencing rule used. These values are the same as shown in Fig. 3 for $N = 30$. Fig. 4 also shows ranges of values for p_1 for which the optimal throughput is unchanged for three variations of the linear model. Case 1 is the linear model including FCFS constraints (6LA) and (6LB). The same optimal throughput can be achieved when p_1 is fixed at any value from 0.50 to 0.89. In Case 2, constraints (6LA) and (6LB) with their underlying sequencing assumption were dropped. The allowable range (0.08, 0.98) for p_1 is considerably wider. Case 3

approximates the nonpreemptive SPT discipline with the addition of two preemptive priority constraints: $z_{1s2s} = 0, s = 1, 4$. These stipulate that at stations s =1 and 4, $W_{1s}(t) = 0$ whenever $X_{2s}(t) > 0$ (see Kumar and Kumar 1994). These constraints severely restrict the inventory mix so that, as in the SPT simulation, nearly all the WIP in the system is comprised of type 1 jobs. When these constraints are included, a vary narrow range (0.977, 0.983) for p_1 is obtained, similar to the p_1 observed in simulation with nonpreemptive SPT sequencing.

***** Figure 4 Here *****

The wide ranges found in the parametric analysis of Cases 1 and 2 suggest that the linear model alone is not a reliable predictor of the WIP mix. Put another way, unless the restrictive preemptive SPT constraints are included, the same maximum throughput can be obtained in the linear model for a wide variety of product mixes. The nonlinear constraints (6NA) and (6NB) therefore are necessary to obtain precise estimates of the WIP mix.

4.2 Example 2.

The second example system illustrates how the results of the single chain planning model can be used in a multiple chain control policy that specifically limits the WIP for each product. This system has five products and ten stations with routings and mean processing times (in minutes) as shown in Table 3. Note that product type 2 visits station 4 twice, with a different expected processing time on each visit. This routing is modeled by using an artificial class 6 to represent class 2 from its first visit to station 4 onward, i.e., $q_{2467} = 1$, so that $P = 5$ product types while $R = 6$ classes. The proportion of type 2 jobs in the WIP mix is calculated as $p_2(N) = \sum_{r=2,6} \sum_{s} \sum_{j} z_{jsrs}/N$.

4.2.1 Throughput Bounds

For the product mix vector $\alpha = (0.2, 0.2, 0.2, 0.2, 0.2)$, Fig. 5 displays simulation results using FCFS sequencing with the bounds obtained by the nonlinear model for different values of N. The

Product	Routing	Processing Times
1	1, 2, 4, 9, 8, 10	8.0, 6.0, 22.8, 8.0, 9.2, 4.0
2	1, 2, 4, 7, 9, 4, 6, 10	8.0, 6.0, 9.2, 12.4, 8.0, 7.6, 14.0, 4.0
3	1, 2, 7, 9, 6, 10	6.0, 4.5, 23.4, 7.2, 9.6, 3.0
4	1, 2, 3, 5, 9, 6	8.0, 6.0, 17.2, 20.4, 6.8, 26.0
5	1, 2, 4, 8, 10	8.0, 6.0, 26.4, 9.2, 4

Table 3. Routings and mean processing times for Example 2.

simulation yields mean throughputs together with their respective 90 percent confidence intervals. For the batch means method of simulation output analysis, throughputs of 30 batches were obtained after a warmup period that was three hundred times as long as each batch. Each batch contained approximately 75 job completions. The lower bounds obtained by minimizing the NLP, $\Theta_{\mathrm{LB}}^{\mathsf{S};\mathrm{NLP}}(N)$, are labeled by their percent difference from their respective upper bounds, $\Theta_{\mathrm{UB}}^{\mathsf{S};\mathrm{NLP}}(N)$. As N becomes large, the distance between the upper and lower bounds becomes smaller than the simulation confidence interval.

*****Figure 5 Here*****

The lower bound on throughput for $N = 90$ shown in Fig. 5 actually was obtained by averaging corresponding values for $N = 89$ and $N = 91$. This represents the only instance in numerous tests on many examples in which minimizing the nonlinear model appears to have resulted in a local, not a global, optimum. However, the anomaly was detected and overcome easily by testing for consistency with adjacent values of N.

For this example, solving the nonlinear model requires on the order of an hour using LINGO on a 200 MHz Pentium, while solving the linear model takes from one to five minutes. These long computation times, which we believe to result partially from subtle redundancies among constraints (2), (4), and (5), point to the need to use both linear and nonlinear models. An application of the method of Section 3.5 to find N' might proceed as follows. The ratio of lower to upper bounds for $N_0 = 20$ from the nonlinear model is $\beta_0 = 0.912$. Since this is lower than $\beta = 0.95$,

N		WIP Mix	Distance	Throughput
30	NLP Min	$(0.215, 0.428, 0.068, 0.079, 0.210)$	0.015	4.30
	NLP Max	$(0.205, 0.428, 0.079, 0.089, 0.198)$	–	4.55
	Simulation	$(0.207, 0.465, 0.077, 0.088, 0.163)$	0.036	4.47
60	NLP Min	$(0.231, 0.462, 0.037, 0.043, 0.228)$	0.014	4.44
	NLP Max	$(0.236, 0.473, 0.028, 0.032, 0.232)$	–	4.55
	Simulation	$(0.165, 0.550, 0.040, 0.043, 0.201)$	0.078	4.51
90	NLP Min	$(0.237, 0.474, 0.025, 0.029, 0.235)$	0.011	4.48
	NLP Max	$(0.242, 0.482, 0.018, 0.019, 0.239)$	–	4.55
	Simulation	$(0.342, 0.440, 0.030, 0.033, 0.156)$	0.097	4.53

Table 4. Comparison of WIP mixes found by the nonlinear model with FCFS simulation results, along with throughput in products per hour, for an equal product mix in Example 2.

we use the linear model to quickly explore bounds for $N > 20$. We find that increasing N from 33 to 34 reduces $\Theta_{\text{UB}}^{\text{S;LP}}(N) - \Theta_{\text{LB}}^{\text{S;LP}}(N)$ by just 0.49 percent. Returning to the nonlinear model, the ratio $\Theta_{\text{LB}}^{\text{S;NLP}}(33)/\Theta_{\text{UB}}^{\text{S;NLP}}(33)$ is 0.953. Of course, further research is required to establish a general relationship between the differences between bounds from the linear model and the ratios of bounds from the nonlinear model.

4.2.2 WIP Mix

As in Example 1, the nonlinear model provides good estimates of the mix of work in process. Table 4 shows the WIP mix vectors predicted by minimizing and maximizing the nonlinear model compared with those found in the FCFS simulation for different values of N. The fourth column shows the Euclidean distances of the WIP mix vectors from the nonlinear maximum, normalized so that the maximum possible distance equals one. Also, as the distance between the NLP-maximizing WIP mixes $p\,(30)$ and $p\,(60)$ is 0.100 while that between $p\,(60)$ and $p\,(90)$ is only 0.021, the WIP mix is fairly stable as N increases.

The WIP mix identified by maximizing the nonlinear model is the mix of work in the system that maximizes throughput, balanced by the single chain release policy to match the desired product mix. This WIP mix should be used in the multiple chain CONWIP control policy. To illustrate, consider the optimal WIP mix $p(30) = (0.205, 0.428, 0.079, 0.089, 0.198)$ that is sug-

Type	NLP Max $6,13,2,3,6$	Sim. Mix $6,14,2,3,5$	From 4 to 1 $7,13,2,2,6$	From 4 to 2 $6,14,2,2,6$	From 4 to 3 $6,13,3,2,6$	From 4 to 5 $6,13,2,2,7$	Equal $6,6,6,6,6$
1	0.91 (0.028)	0.90 (0.028)	0.97 (0.020)	0.83 (0.026)	0.94 (0.026)	0.82 (0.020)	1.03 (0.028)
2	0.94 (0.029)	1.00 (0.042)	0.90 (0.028)	0.94 (0.030)	0.97 (0.038)	0.87 (0.024)	0.43 (0.017)
3	0.93 (0.042)	0.95 (0.038)	1.12 (0.027)	1.16 (0.033)	1.47 (0.042)	1.14 (0.033)	1.55 (0.061)
4	1.14 (0.033)	1.15 (0.034)	0.96 (0.017)	0.98 (0.027)	0.92 (0.024)	0.97 (0.027)	1.50 (0.041)
5	0.95 (0.027)	0.79 (0.024)	0.86 (0.017)	0.85 (0.026)	0.97 (0.029)	0.97 (0.026)	1.09 (0.033)
Total	4.86 (0.084)	4.79 (0.096)	4.81 (0.063)	4.77 (0.063)	5.27 (0.069)	4.75 (0.067)	5.61 (0.111)
$\|\gamma - \alpha\|$	0.028	0.040	0.030	0.039	0.062	0.036	0.114

Table 5. Mean throughput in products per hour (with standard errors) found by simulation using a multiple chain control policy for different allocations of 30 kanbans.

gested by maximizing the NLP with $N = 30$. The system population limits for individual products in the multiple chain policy are obtained by multiplying the WIP mix vector by N, i.e., $30(0.205, 0.428, 0.079, 0.089, 0.198) = (6, 13, 2, 3, 6)$. Likewise the product population limit vector suggested by the single chain simulation is $(6, 14, 2, 3, 5)$. Recall that these limits were obtained with the requirement of an equal product mix of 20% for each product.

Table 5 compares means and standard errors of throughput from simulation under the multiple chain control policy for these WIP limit vectors as well as others. Note that the mean total throughput for each vector of WIP limits is higher than that found in the single chain simulation (4.47 per hour as reported in Table 4). The normalized Euclidean distances of the throughput mix vectors from the ideal product mix of $(0.2, 0.2, 0.2, 0.2, 0.2)$ are shown in the last row. Since product type 4 has the highest throughput, the columns labeled "From 4 to x" explore the effect of reallocating one of its kanbans to each other product type. In each case, this reallocation increases the distance from the desired product mix. Finally, the last column shows that naively equating the WIP mix with the desired product mix yields a very inequitable distribution of throughput across the products.

5 Conclusions

Determining individual WIP limits for separate products in a job shop environment is a difficult

stochastic combinatorial problem. We modeled the job shop as a closed queuing network and introduced a systematic procedure to set WIP levels for a given product mix by identifying a total WIP guaranteed to achieve some proportion of the maximum possible throughput and a WIP mix that matches throughput proportions to the product mix. The procedure involves a novel application of mathematical programs that bound throughput.

When individual WIP limits derived from the WIP total and mix are applied in simulation, the product throughputs compare favorably with those obtained from alternative allocations of the same total WIP. In addition, our simulation experiments provided some insight into the impact of sequencing rules on WIP mix and throughput mix. When the system is operated as a single chain queuing network, with throughput mix matching the product mix, sequencing has a relatively small impact on total throughput but it affects the WIP mix significantly.

The results of this work indicate the need for further analytical modeling to understand pull and hybrid push-pull control policies in complex multi-product environments. Our pragmatic application of single-chain analysis for multiple-chain operation highlights an open question: when producing multiple products with limited WIP, should the policy maintain a single WIP level for all products or individual levels for each product? Most of the existing studies rely on simulation and the results appear to depend heavily on the system configuration and differences in product routings. These models, along with improved search procedures for finding WIP levels, await the development of more general and accurate performance evaluation models for kanban-type systems.

Acknowledgement 1 *This work was supported in part by NSF grants DMI-9701403 (Ryan) and DDM-9400146 (Choobineh). Additional support was provided by the Layman Fund at the University of Nebraska-Lincoln.*

References

Bard, J. F. and Golany, B. (1991) Determining the number of kanbans in a multiproduct, multistage production system. *International Journal of Production Research*, **29**(5), 881-895.

Baynat, B. and Dallery, Y. (1996) A product-form approximation method for general closed queueing networks with several classes of customers. *Performance Evaluation*, **24**, 165-188.

Bertsimas, D., Paschalidis I. Ch. and Tsitsiklis, J. N. (1994) Optimization of multiclass queuing networks: polyhedral and nonlinear characterizations of achievable performance. *The Annals of Applied Probability*, **4**(1), 43-75.

Buzacott, J. A. and Shanthikumar, J. G. (1993) *Stochastic Models of Manufacturing Systems*, Prentice Hall, Englewood Cliffs, NJ.

Chevalier, P. B. and Wein, L. M. (1993) Scheduling networks of queues: heavy traffic analysis of a multistation closed network. *Operations Research*, **41**(4), 743-758.

Choobineh, F. and Sowrirajan, S. (1996) Capacitated - constant work in process (C-CONWIP): a job shop control system. Technical Report, Industrial and Management Systems Engineering, University of Nebraska, Lincoln, NE 68588-0518.

Dallery, Y. and Frein, Y. (1988) An efficient method to determine the optimal configuration of a flexible manufacturing system. *Annals of Operations Research*, **15**, 207-225.

Dar-El, E. M., Herer, Y. T. and Masin, M. (1999) CONWIP-based production lines with multiple bottlenecks: performance and design implications. *IIE Transactions*, **31**, 99-111.

Duenyas, I. (1994) A simple release policy for networks of queues with controllable inputs. *Operations Research*, **42**(6), 1162-1171.

Golany, B., Dar-El, E. M. and Zeev, N. (1999) Controlling shop floor operations in a multi-family, multi-cell manufacturing environment through constant work-in-process. *IIE Transactions*,

31, 771-781.

Gstettner, S. and Kuhn, H. (1996) Analysis of production control systems kanban and CONWIP. *International Journal of Production Research*, **34**(11) , 3253-3273.

Herer, Y. T. and Masin, M. (1997) Mathematical programming formulation of CONWIP based production lines; and relationships to MRP. *International Journal of Production Research*, **35**(4), 1067-1076.

Hopp, W. J. and Roof, M. L. (1998) Setting WIP levels with statistical throughput control (STC) in CONWIP production lines. *International Journal of Production Research*, **36**(4), 867-882.

Jin, H., Ou, J. and Kumar, P. R. (1997) The throughput of irreducible closed Markovian queueing networks: functional bounds, asymptotic loss, efficiency, and the Harrison-Wein conjectures. *Mathematics of Operations Research*, **22**(4), 886-920.

Kerola, T. (1986) The composite bound method for computing throughput bounds in multiple class environments. *Performance Evaluation*, **6**, 1-9.

Kouvelis, P. and Kiran, A. S. (1991) The plant layout problem in automated manufacturing systems. *Annals of Operations Research*, **26**, 397-412.

Kouvelis, P. and Lee, H. L. (1995) An improved algorithm for optimizing a closed queuing network model of a flexible manufacturing system. *IIE Transactions*, **27**, 1-8.

Krishnamurthy, A. and Suri, R. (2000) Re-examining the performance of push, pull and hybrid material control strategies for multi-product flexible manufacturing systems. Technical Report, Center for Quick Reponse Manufacturing, University of Wisconsin-Madison, Madison, WI 53706-1572.

Kumar, S. and Kumar, P. R. (1994) "Performance Bounds for Queuing Networks and Scheduling Policies," *IEEE Trans. Automatic Control*, **39**(8), 1600-1611.

Lee, H. F., Srinivasan, M. M. and Yano, C. A. (1989) The optimal configuration and workload allocation problem in flexible manufacturing systems. *Proceedings of the Third ORSA/TIMS Conference on Flexible Manufacturing Systems*, 85-90.

Luh, P. B., Zhou, X. and Tomastik, R. N. (2000) An effective method to reduce inventory in job shops. *IEEE Trans. Robotics and Automation* **16**(4), *420-424.*

Reiser, M. and Kobayashi, H. (1975) Queuing networks with multiple closed chains: theory and computational algorithms. *IBM Journal of Research and Development*, **19**, 283-294.

Reiser, M. and Lavenberg, S. S. (1980) Mean-value analysis of closed multichain queuing networks. *Journal of the ACM*, **27**, 313-322.

Ryan, S. M., Baynat, B. and Choobineh, F. (2000) Determining inventory levels in a CONWIP controlled job shop. *IIE Transactions*, **32**(2), 105-114.

Schweitzer, P. J., Seidmann, A. and Shalev-Oren, S. (1986) The correction terms in approximate mean value analysis. *Operations Research Letters*, **4**(5), 197-200.

Shanthikumar, J. G. and Yao, D. D. (1988) Second-order properties of the throughput of a closed queuing network. *Mathematics of Operations Research*, **13**(3), 524-534.

Spearman, M., Woodruff, D. and Hopp, W. (1990) CONWIP: A pull alternative to kanban. *International J. Production Research*, **28**, 879-894.

Spearman, M. and Zazanis, M. (1992) Push and pull production systems: issues and comparisons. *Operations Research*, **40**, 521-532.

Spearman, M. (1992) Customer service in pull production systems. *Operations Research*, **40**, 948-958.

Suri, R. (1985) A concept of monotonicity and its characterization for closed queuing networks. *Operations Research*, **33**, 606-624.

Suri, R. (1998) *Quick Response Manufacturing*, Productivity Press, Portland, OR.

Tayur, S. (1993) Structural properties and a heuristic for kanban-controlled serial lines. *Management Science*, **39**(11), 1347-1368.

Vinod, B. and Solberg, J. J. **(**1985) Optimal design of flexible manufacturing systems. *International Journal of Production Research*, **23**(6), 1141-51.

Zhou, X., Luh, P. B. and Tomastik, R. N. (2000) The performance of a new material control and replenishment system: a simulation and comparative study. *Proceedings of the Quick Response Manufacturing 2000 Conference*, 807-826.

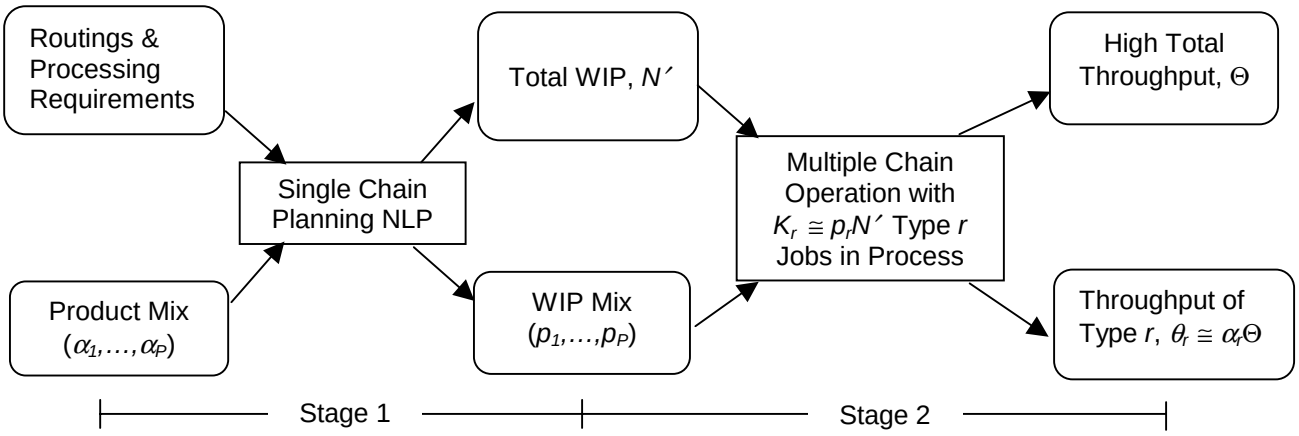
Routings & Processing Requirements
Product Mix ($\alpha_1,\ldots,\alpha_P$)
Single Chain Planning NLP
Total WIP, N'
WIP Mix ($p_1,\ldots,p_P$)
Multiple Chain Operation with $K_r \cong p_r N'$ Type r Jobs in Process
High Total Throughput, Θ
Throughput of Type r, $\theta_r \cong \alpha_r \Theta$
Stage 1
Stage 2

Figure 1.

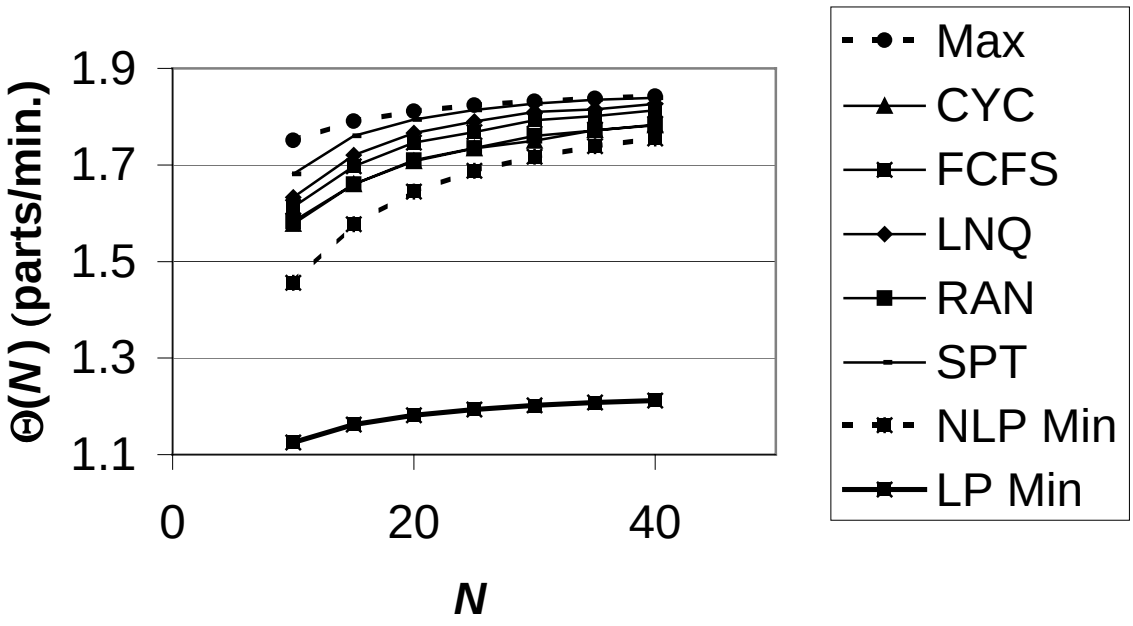
1.9
1.7
1.5
1.3
1.1
Θ(N) (parts/min.)
0
20
40
N
Max
CYC
FCFS
LNQ
RAN
SPT
NLP Min
LP Min

Figure 2.

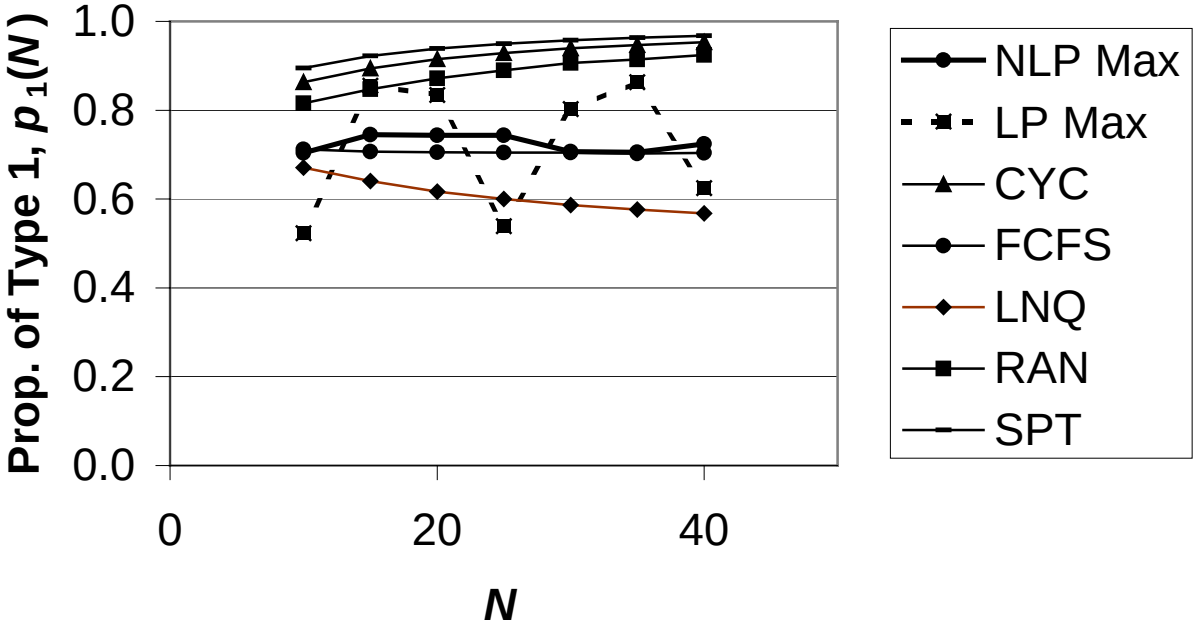
Prop. of Type 1, $p_1(N)$
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
N
NLP Max
LP Max
CYC
FCFS
LNQ
RAN
SPT

Figure 3.

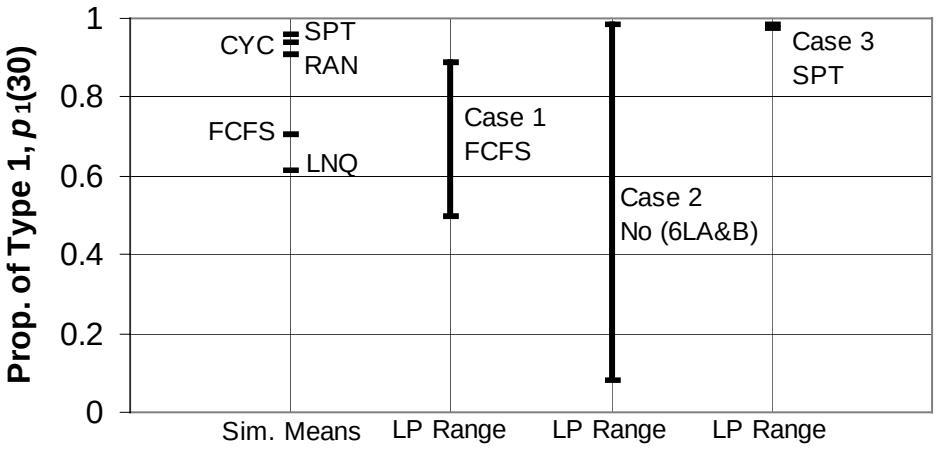
Prop. of Type 1, $p_1(30)$
1
0.8
0.6
0.4
0.2
0
CYC
SPT
RAN
FCFS
LNQ
Case 1
FCFS
Case 2
No (6LA&B)
Case 3
SPT
Sim. Means
LP Range
LP Range
LP Range

Figure 4.

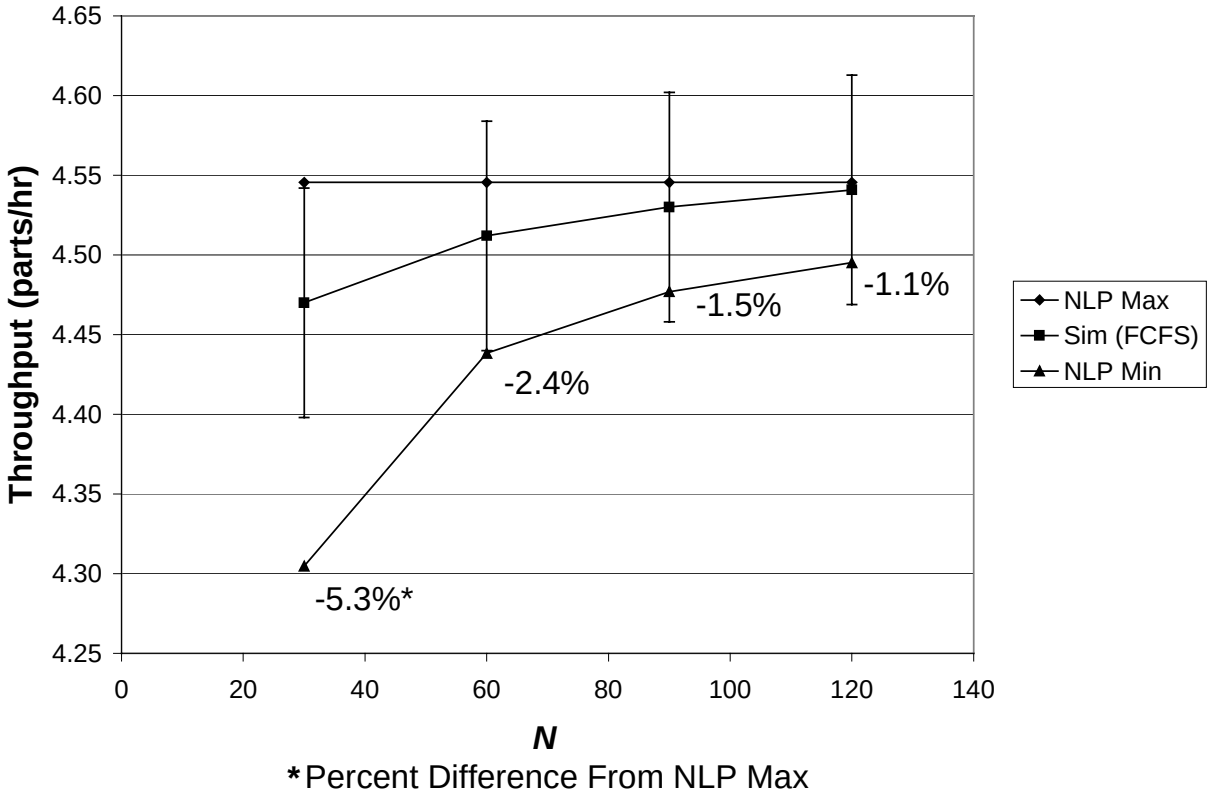
4.65
4.60
4.55
4.50
4.45
4.40
4.35
4.30
4.25
Throughput (parts/hr)
0
20
40
60
80
100
120
140
N
-5.3%*
-2.4%
-1.5%
-1.1%
NLP Max
Sim (FCFS)
NLP Min
*Percent Difference From NLP Max

Figure 5.

Figure Captions

Figure 1. The proposed two stage procedure for planning WIP total and mix using a single chain model and applying the results in a multiple chain operational control.

Figure 2. Throughput predicted by the model and observed in simulation for Example 1 with α = (0.3, 0.7) and μ = 3.

Figure 3. Proportion of type 1 jobs predicted by maximizing the linear and nonlinear models and observed in simulation, for Example 1 with α = (0.7, 0.3) and μ = 2.

Figure 4. Ranges of the proportion of type 1 parts that allow maximum throughput in the linear model to stay the same under different sequencing rules for Example 1 with α = (0.7, 0.3), μ = 2 and N = 30.

Figure 5. NLP throughput bounds compared with simulation confidence intervals for Example 2 with an equal product mix.

Biographical Sketches

Sarah M. Ryan is an Associate Professor of Industrial and Manufacturing Systems Engineering at Iowa State University, having taught previously at the Universities of Pittsburgh and Nebraska. She holds a B.S. in Systems Engineering from The University of Virginia and M.S.E. and Ph.D. degrees in Industrial and Operations Engineering from The University of Michigan. Her research interests are in estimation and planning of manufacturing and service systems and sequential decision making. She is a senior member of IIE and a member of INFORMS and ASEE.

F. Fred Choobineh is a Professor of Industrial and Management Systems Engineering at the University of Nebraska-Lincoln. He received B.S.E.E., M.S.I.E. and Ph.D. degrees from Iowa State University. He is a licensed Professional Engineer in the State of Nebraska and serves as the chair of the State's Board of Engineers and Architects. His research interests are related to the design and control of manufacturing systems and use of approximate reasoning techniques such as Rough Sets and Evidence Theory in decision making. He is a fellow of IIE and a member of IEEE, ASEE and INFORMS.