Throughput Analysis in Manufacturing Networds

by
Gabriel R. Bitran
Deb Sarkar

WP #3230-90-MSA December 1990

Throughput Analysis in Manufacturing Networks

G. R. Bitran*
Sloan School of Management
MIT
Cambridge, Mass. 02139

D. Sarkar
AT&T Bell Laboratories
Crawfords Corner Road
Holmdel, NJ 07733

ABSTRACT

The throughput of a plant is a measure of major importance when assessing its ability to compete successfully in the market place. Managers often rely on changes in capacity and process improvements as two major factors that impact throughput. In open networks of queues the optimal allocation of resources to these two factors is difficult to determine without the support of appropriate mathematical models. In this paper we attempt to quantify the tradeoffs between capacity and process improvements, through variance reductions, and throughput.

We consider multiproduct manufacturing systems modeled by open networks of queues and formulate the throughput characterization (TC) and variability reduction (VR) problems as nonlinear programs. These formulations are based on the decomposition approach for estimating the work-in-progress in open queueing networks. The TC (under some mild assumptions) and VR programs are shown to be equivalent to convex programming problems. We present and analyze greedy-type heuristics that facilitate the derivation of tradeoff curves for these problems. The implications of the assumptions made in developing the heuristic for the TC problem are also examined.

November 1990

* Research partially supported by the Leaders for Manufacturing Program.

Throughput Analysis in Manufacturing Networks

G. R. Bitran
Sloan School of Management
MIT
Cambridge, Mass. 02139

D. Sarkar
AT&T Bell Laboratories
Crawfords Corner Road
Holmdel, NJ 07733

1. Introduction

The throughput of a plant is a measure of major importance when assessing its ability to compete successfully in the market place. Managers often rely on changes in capacity and process improvements as two major factors that impact throughput. In open networks of queues the optimal allocation of resources to these two factors is difficult to determine without the support of appropriate mathematical models. In this paper we attempt to quantify the tradeoffs between capacity and process improvements, through variance reductions, and throughput.

The importance of the relationships between performance criteria such as work-in-progress (WIP), lead times, throughput, manufacturing costs, operation and capital investments has been emphasized by many authors [Skinner [1974], Hayes and Wheelwright [1984], Bitran and Tirupati [1989a], Boxma et. al. [1990]). These relationships are typically expressed in terms of tradeoff curves. These curves allow managers to examine their firm's strategic objectives (high volume producer, fast supplier, low cost producer, ...) against the investments required to support such objectives. Bitran and Tirupati, for example, studied the tradeoff between WIP and capacity (expressed in terms of service rates) for a production facility that manufactures semiconductor devices. The analysis (and also that of Boxma et. al. [1990]) assumes that the throughput, product mix and technology are given. They, however, comment that the "... concept of tradeoff curves is also useful in examining the impact of changes in throughput, product mix and technology on WIP and lead times."

Most papers to date study the impact of capacity changes on WIP. However, process improvements such as variability reductions also lead to lower WIP and higher effective throughput. The *qualitative* importance of continuous process improvements to attain manufacturing excellence has been

brought out by many authors (Imai [1986], Hall [1983], Schonberger [1982]). In this paper we study the quantitative impact of variability reductions on WIP and throughput. Sarkar and Zangwill [1988] provide exact analysis on variability reduction and throughput characterization in a cyclic production system. The methodology of this paper, on the other hand, is based on approximate formulae and is applicable to more general manufacturing environments.

We consider a production system that can be modelled as an open network of queues with different product classes. That is, the production system consists of multiple workstations through which products flow as per a markovian routing matrix. Each workstation is modelled as a GI/G/1 queue and parametric decomposition methods are used to estimate the queue lengths at each station. The assumption that the queueing network is comprised of single server stations is not essential. The methodology is easily extended to multiserver stations. This requires the use of appropriate approximations for the expected queue length at each station (Whitt [1983a, 1985], Bitran and Tirupati [1989b]).

The throughput characterization problem addresses the issue of service rate (capacity) assignment to workstations in order to meet a target throughput level so that the initial WIP is not exceeded and the cost of capacity assignment is minimum. In order to solve the throughput characterization problem, we assume, initially, that the squared coefficient of variations (scvs) of the interarrival times and the processing times at each of the workstations are independent of the capacity changes. With this condition, a finite procedure for constructing the throughput-capacity tradeoff curve is developed. While this is reasonable in networks with a large number of products, it is not difficult to construct examples in which this assumption does not hold. The implication of this assumption and the development of tradeoff curves without this condition are also discussed in this paper. Specifically, we propose an iterative scheme for the general case and establish sufficient conditions for its convergence.

To study the impact on WIP and throughput, of reducing variabilities associated with external arrivals and processing at each of the workstations an optimization problem is formulated. The formulation attempts to prioritize different process improvement options in order to meet a WIP target. We present an efficient heuristic to construct the WIP-variability tradeoff curve based on the formulation and discuss its properties. We also show how variability reduction leads to increased effective throughput.

This paper is organized as follows. In Section 2, we describe the model and related notation. The impact of variability reductions on WIP is analyzed in Section 3. The throughput-capacity tradeoff is described in detail in Section 4. The assumption that the scvs of interarrival and processing times are independent of capacity changes is examined in Section 5. Section 6 contains the conclusions.

2. *Model and Notation*

The production system has J workstations. Each workstation is modelled as a single server queue with general arrival and service time distributions. N product types are produced by the network. After visiting the first station the products follow a markovian routing within the network. We treat the different product types as one aggregate product with an aggregate arrival rate λ_j at workstation j. The *aggregate* service time at workstation j has mean $\frac{1}{\mu_j}$ and variance σ_j^2. Computations of aggregate arrival and service time parameters are discussed in Whitt [1983a].

The motivation for considering such model is described in depth in Bitran and Tirupati [1989a] and other papers cited there. Since exact results for performance measures do not exist for open network models, we employ the parametric decomposition approach to derive estimates. This method has been used by many authors to analyze general networks (e.g., Whitt [1983a, b], Shanthikumar and Buzacott [1981], Buzacott and Shanthikumar [1985], Bitran and Tirupati [1988]).

The parametric decomposition approach assumes that the product characteristics (mean and scv of the interarrival times and the markovian routing for each aggregate product) are specified. Likewise, the mean and scv of processing times at each station are known.

We associate an average monetary value (v_j) with each aggregate job at station j to compute the value of WIP. The average WIP at stations j is the product of v_j and the mean number of jobs at the station (L_j). The v_j's can be estimated in several ways. For example, they may be determined as a weighted sum of per job WIP value for each product family. The weights are proportional to the product arrival rate and the average waiting time for each product family. Albin [1986] describes an approximate procedure to compute product family waiting times. The notation used in this paper is defined below:

J: number of stations in the network.

λ_j: net arrival rate at station j.

μ_j: capacity (service rate) at station j; μ_j^I is the initial capacity.

ρ_j: average utilization at station j; $\rho_j = \lambda_j/\mu_j$. ρ_j^I is the initial utilization.

L_j: expected number of jobs at station j; L_j' is the derivative of L_j with respect to μ_j.

v_j: average WIP value for each job at station j.

$F_j(\mu_j)$: investment function (cost associated with capacity μ_j) at station j; $F_j'(\mu_j)$ is the derivative of $F_j(\mu_j)$ with respect to μ_j. ($F_j(\mu_j^I) \triangleq 0$).

W: total value of WIP in the network; W^I is the initial value and W^T denotes the target value.

scv: squared coefficient of variation; the scv of a random variable is the ratio of its variance to the square of its mean.

ca_j: scv of the interarrival time at station j.

cs_j: scv of service times at station j.

The decomposition approach provides an approximation for the mean number of jobs at a station j. We use the formula due to Kraemer and Langenbach-Belz [1976] and its modification by Whitt [1983a]. This approximation is given by

$$L_j = \rho_j + \frac{\rho_j^2}{2(1-\rho_j)}(ca_j + cs_j)g(\rho_j, ca_j, cs_j) \tag{2.1}$$

where

$$g(\rho_j, ca_j, cs_j) = \begin{cases} \exp\left\{\dfrac{-2(1-ca_j)}{3(ca_j+cs_j)}\dfrac{(1-\rho_j)}{\rho_j}\right\} & \text{if } ca_j \leqq 1 \\ 1 & \text{otherwise} \end{cases}$$

and the *ca*s are determined by solving the following two systems of equations.

$$\lambda_i = \lambda_{0i} + \sum_{j=1}^{J} \lambda_j r_{ji},\ i = 1, 2, ..., J \tag{2.2}$$

$$\lambda_i ca_i - \sum_{j=1}^{J} \left\{\lambda_j (1-\rho_j^2) r_{ji}^2 ca_j\right\}$$

$$= \lambda_{0i}\, ca_i^0 + \sum_{j=1}^{J} \left\{\lambda_j r_{ji}(\rho_j^2 r_{ji} cs_j + 1 - r_{ji})\right\} \tag{2.3}$$

$$i = 1, 2, ..., J$$

with

R = routing matrix $[r_{ij}]$: r_{ij} is the probability that a job visiting station i will visit j after completion of service at i.

λ_{0i} = total external arrival rate at station i.

$ca_i^0 \equiv$ scv of the external arrival interval at station i.

Note that λ_{0i} and ca_i^0 are related to the external product arrival parameters as follows:

$$\lambda_{0i} = \sum_{p=1}^{P} \lambda_p^e \xi_{ip}$$

$$ca_i^0 = \sum_{p=1}^{P} ca_p^e \lambda_p^e \xi_{ip} / \lambda_{0i}$$

where λ_p^e, ca_p^e are *respectively the external* arrival rate and scv of interarrival times of product p, and

ξ_{ip} = 1 if product p arrives at station i first

= 0 otherwise

In our analysis we take λ_{0i} and ca_i^0 as given.

3. Throughput and Variability Reduction

The reduction of (external) interarrival or processing time variabilities at the workstations can be part of KAIZEN or continuous, incremental improvement approach (Imai [1986]). In this section we examine this important approach by quantifying the results obtained through reducing variabilities.

The arrival and service variabilities encompass several different elements. The arrival variability, for example, may include variations in market demand, order changes, product design changes, unavailability of parts and so on. The service variability, on the other hand, may include factors such as service time variations due to aggregation of products, disruptions (machine breakdowns), operator experience and training etc. This section's focus is on reducing product and process variabilities related to disruptions, unavailability of parts, release of products to shop floors and others. It is possible that certain types of variability reductions lead to changes in the routing of products. For example, better process control may lead to lower fraction of products visiting a rework station. In our analysis, in this section, we have assumed that the variability reductions do not have any impact on the routing probabilities of products. The methodology of this section is extended in Section 3.1 to analyze this case. Expressing equation (2.3) in terms of variance of external interarrival time at station j, δ_j^2, and variance of processing time at node j, σ_j^2, we have

$$\lambda_i ca_i - \sum_{j=1}^{J} \lambda_j r_{ji}^2 (1 - \rho_j^2) ca_j = \lambda_{0i}^3 \delta_i^2 + \sum_{j=1}^{J} \lambda_j r_{ji} (\rho_j^2 r_{ji} \sigma_j^2 \mu_j^2 + 1 - r_{ji})$$

$$i = 1, 2, ..., J \qquad (3.1)$$

The δs are defined for nodes with external arrivals. Let E be the set of nodes that have external arrivals.

We define $\delta_i = 0$ for $i \notin E$. (3.1) is a Leontief system and it is easy to see that

$$\frac{\partial ca_i}{\partial(\delta_k^2)} \geqq 0 \text{ and } \frac{\partial ca_i}{\partial(\sigma_j^2)} \geqq 0 \text{ for } i, j = 1, 2, ...,J \text{ and } k \in E \quad \text{(with strict inequality for at least one } i\text{)}$$

So, using (2.1), WIP tends to decrease as δ and/or σ are reduced. Alternately, suppose the initial WIP is W^I and some of the variabilities are reduced. Let ca_i', $i = 1, 2, ...,J$ denote the solution to (3.1) at the reduced value of the variabilities. Let $(1 + \beta)\lambda_i^I$ be the mean arrival rate at each station i ($i = 1, 2, ...,J$), if variances were reduced, but total WIP was kept the same. Then the value of $(1 + \beta)$ from the following equation provides an estimate of the increase in effective throughput due to variability reductions:

$$\sum_{j=1}^{J} v_j L_j((1 + \beta)\lambda_j^I, \mu_j^I, ca_j', cs_j) = W^I \tag{3.2}$$

Since

$$W^I = \sum_{j=1}^{J} v_j L_j(\lambda_j^I, \mu_j^I, ca_j^I, cs_j^I) \text{ and } ca_j' \leqq ca_j^I \text{ (with strict inequality for at least one } j\text{)},$$

$\beta > 0$ and the effective throughput increases as a result of variability reductions. Finally, by setting all δ^2 and σ^2 to zero in equation (3.1) and using the resultant cas in the left hand side of (3.2), we obtain an upper bound on the WIP reduction or throughput increase achievable through variability reductions. Also, note that $1 + \beta < \min_j \frac{1}{\rho_j^I}$.

Example: We now present an example to highlight the tradeoffs between different variability reductions and show that these can lead to substantial WIP reduction. Consider a slight variation of the real life network model (Bitran and Tirupati [1988, 1989a]) of a semiconductor production facility. The facility is represented by 14 machine stations and processes jobs of 10 product families. Station 14 is a rework station associated with station 9. The network characteristics are described, in detail, in Appendix 3.

QNA3.3 software (Segal and Whitt [1988]) was used to obtain the results reported in this example.

When the arrival rates of all 10 product types are taken to be equal to 0.1, the total WIP is 33.19. The utilization of each of the machines corresponding to this arrival rate is shown in Appendix 3. If the service time variability of station 9 (which has the highest utilization) is completely eliminated, the WIP is reduced by 11.8% to 29.26. Elimination of all service time variabilities, on the other hand, results in 48.8% reduction in the WIP.

While the above reductions are significant, the magnitude could be dramatic at even higher utilization. For example, consider the same network with arrival rate = 0.1062 for all products. The total WIP corresponding to this arrival rate is 219.75. Now if the service time variability of station 9 is completely eliminated, the new WIP is reduced by 66.3% to 74.04.

Only node 1, in this example, has external arrivals. Since node 1 has relatively low utilization we expect that the elimination of all arrival time variabilities would not lead to much improvement in the WIP. Indeed this is the case. Elimination of all arrival time variability reduces WIP by 3.8%. However, were the utilization of station 1 higher, the *impact* of arrival time variability reduction would be greater. For example, if the mean service time of station 1 is 0.95 then reduction of all arrival variability would imply WIP reduction of 10.85%.

Since $ca_1^0 > cs_1$, in this example, and $\frac{\partial L_1}{\partial ca_1^0} > \frac{\partial L_1}{\partial cs_1}$ (can be verified by differentiating (2.1)), one might think that reduction in arrival variability would lead to greater reduction in WIP than an equivalent reduction in the service time variability of station 1. But this may not be the case since the cas of other nodes are also affected by these reductions and total effects need to be compared. For example, if service time variability of node 1 is completely eliminated (at arrival rate = 0.1062), the WIP reduction is 2.4%, which is higher than the 1.8% reduction obtained through the arrival variability reduction. Note that, in this example, the scv of arrival and service times are respectively 0.375 and 0.333.

In practice, often, variability reductions are accompanied by some costs. The costs could stem from training labor, making labor available on time, controlling releases to the floor or process improvements. The problem then is how to prioritize the cost expenditures. We next present a formulation

that addresses this issue. In this problem a WIP cost target is to be achieved by reducing variabilities. The objective is to minimize the investment costs associated with reductions of variabilities. The problem is written as

$$(VR): \qquad \min \sum_{j=1}^{J} G_j(\sigma_j^2) + \sum_{i \in E} H_i(\delta_i^2) \tag{3.3a}$$

subject to (2.2), (3.1)

$$\sum_{j=1}^{J} v_j L_j \leqq W^T \tag{3.3b}$$

$$0 \leqq \sigma_j^2 \leqq (\sigma_j^I)^2,\ 0 \leqq \delta_i^2 \leqq (\delta_i^I)^2 \tag{3.3c}$$

(3.3a) denotes the investment costs while (3.3b) is the WIP cost target constraint. $G(\cdot)$ and $H(\cdot)$s denote the investments required to bring the respective variates to the argument values. We assume that the functions $G(\cdot)$ and $H(\cdot)$ are convex and differentiable. The convexity assumption is reasonable since it models situations where variability reductions are achieved by employing cheaper options initially. The differentiability assumption is not crucial to our analysis. This assumption, however, makes the analysis simpler. The methodology holds for cases where variabilities are reduced in discrete amounts (as an illustration see the discrete capacity option problem in Bitran and Tirupati [1989b]).

From (3.1) it is easy to see that the *cas* are linear functions of δ^2 and σ^2. Also, since each L_j is jointly convex in ca_j and cs_j (Appendix 1), using theorem 6.9 in Avriel [1976], each L_j is convex in $\{\delta^2, \sigma^2\}$. The problem (VR) is thus a convex programming problem. We now present a greedy heuristic to solve (VR) as it facilitates the construction of the WIP-variability tradeoff curve. Note that the throughput-variability tradeoff curve is easily derivable from the WIP-variability curve using equation (3.2).

At each iteration of the heuristic, we evaluate two priority indices, PI_i' and PI_j'' as follows:

$$PI_i' = -\frac{\frac{\partial W}{\partial \delta_i^2}}{\frac{\partial H_i}{\partial \delta_i^2}}, \quad i \in E \tag{3.4}$$

$$PI_j'' = -\frac{\frac{\partial W}{\partial \sigma_j^2}}{\frac{\partial G_j}{\partial \sigma_j^2}}, \quad j = 1, 2, ..., J \tag{3.5}$$

(Note: $W \triangleq \sum_{j=1}^{J} v_j L_j$)

Computations of PI_i' and PI_j'' are relatively straightforward and discussed in Appendix 2. The station with the highest PI (be it PI' or PI'') is targeted for variability reduction at that stage. For example, if PI_k' yields the maximum value, then δ_k^2 is reduced by a predetermined value Δ and the priority indices are recomputed. (In our presentation we have assumed Δ to be the same for both δ^2 and σ^2. The methodology, however, holds if δ^2 and σ^2 are reduced in different step sizes.)

The process is repeated until the desired WIP cost target W^T is achieved or no further improvement in variabilities can be made. In the latter case, there is no feasible solution and the heuristic terminates.

Heuristic

Step 1: Initialization: $\delta_i^2 = (\delta_i^I)^2$, $\sigma_j^2 = (\sigma_j^I)^2$ and the corresponding $WIP = W^I$. Compute the priority indices PI_i' and PI_j'' for $i \in E$ and $j = 1, ..., J$.

Step 2: If $WIP \leq W^T$, or (δ_j^2 and $\sigma_j^2 < \Delta$) stop; else, go to step 3.

Step 3: Let

$$j_1^* = argmax\ \{PI'_j,\ j \in E\} \text{ and}$$

$$j_2^* = argmax\ \{PI''_j,\ j = 1,\ 2,\ \ldots J\}$$

If $PI'_{j_1^*} \geq PI''_{j_2^*}$, set $\delta^2_{j_1^*} = \delta^2_{j_1^*} - \Delta$.

If $PI'_{j_1^*} < PI''_{j_2^*}$, set $\sigma^2_{j_2^*} = \sigma^2_{j_2^*} - \Delta$.

Update *ca*, *cs*, PI', PI'' and *WIP*; go to step 2.

Again, the heuristic converges in a finite number of steps because the variance is reduced by a fixed amount at each iteration. Also, each iteration of the heuristic gives a point on the WIP-variability tradeoff curve. When Δ is small, the tradeoff curve generated by the heuristic will be fairly close to the exact one.

3.1 Variability Reductions and Changes in Routing Probabilities

We have so far assumed that the variability reductions do not impact the routing probabilities of the products in the network. Better product designs, process control and/or other quality improvements, however, may lead to fewer visits to rework centers and hence, can have an impact on the routing of products. In this section we examine a model in which the probability of visiting the rework stations depend on the service time variance. The analysis presented here can, however, be applied to quantify tradeoffs in other quality improvement investments that affect the product routing probabilities. Our main objective is to show that the methodology presented thus far applies to this case.

For simplicity of exposition we consider a network with $2J$ nodes. Each station j (j=1, 2, ... J) has a rework station associated with it. A product, after completing service at station j, either goes to its rework station (with probability α_j) or visits station i with probability $(1-\alpha_j)r_{ji}$. The mean and scv of the service time at the rework node of station j are θ_j and $c\theta_j$ respectively. After completing service at the rework node of station j, products visit station i with probability r_{ji}.

When equations (2.2) and (2.3) are examined for this system with rework, its analysis decomposes into two parts. The cas for the J workstations can be obtained first by solving a linear system of J equations. The ca value of station j can then be used to determine the ca value of the rework node

associated with station j. We denote the ca of the rework node j by cr_j. So, the L_j for each station and rework node is known. We note that the λ_j for each station j is not affected by the values of α_js. Our goal is to show that, under reasonable assumptions, each L_j is convex in $\{\sigma^2, \delta^2\}$. This will then imply that the methodology discussed for the (VR) problem is applicable to this model.

The cas for the J stations, after some algebra, can be shown to satisfy the following relations. (The λ_js are given by (2.2)).

$$\lambda_i ca_i - \sum_j \lambda_j r_{ji}^2 \left[1 - \left\{1 - \left[(1-\alpha_j)^2 + \alpha_j^2(1-\rho_j'^2)\right](1-\rho_j^2)\right\}\right] ca_j$$

$$= \lambda_{0i} ca_i^0 + \gamma_i + \sum_j \lambda_j r_{ji} \left[r_{ji}\left\{1 - \left[(1-\alpha_j)^2 + \alpha_j^2(1-\rho_j'^2)\right](1-\rho_j^2)\right\} cs_j + 1 - r_{ji}\right]$$

$$i = 1, 2, \ldots J \tag{3.6}$$

where

$$\rho_i' = \alpha_i \lambda_i \theta_i \tag{3.7}$$

$$\gamma_i = \sum_j \lambda_j r_{ji}^2 \left[2(1-cs_j)\alpha_j - 2(1-cs_j)\alpha_j^2 + (c\theta_j - 1)(\lambda_j\theta_j)^2\alpha_j^3 + (1-cs_j)(\lambda_j\theta_j)^2\alpha_j^4\right] \tag{3.8}$$

Comparing (3.6) to (2.3) we observe that, as far as the computation of the cas is concerned, the α_js have the "equivalent" impact of increasing the mean service time and the external arrival variabilities. To see this, when the $\lambda_{oi}\, ca_i^o$ and ρ_j^2 terms in (2.3) are replaced by $\lambda_{oi} ca_i^o + \gamma_i$ and $r_j^2 \triangleq 1 - ((1 - \alpha_j)^2 + \alpha_j^2 (1 - \rho_j'^2))(1 - \rho_j^2)$, respectively, we obtain equation (3.6). Since $\gamma_i \geq 0$ (see discussion below) and $r_j \geq \rho_j$, (3.6) is equivalent to a system with higher arrival variability and mean service times than the one given by (2.3). The rework nodes, as one would expect, have the impact of increasing the cas of the workstations.

One might also expect that the increase in the cas can be interpreted solely in terms of increased external arrival variabilities at the nodes. But equation (3.6), interestingly, as explained above suggests that, as far as computation of the cas matters, not only the external arrival variability at each node is increased but also the mean capacity at each node is reduced. In the discussion above we have implicitly assumed that the γ_is are non-negative. This is reasonable because θ and $c\theta$ for rework stations, which may involve considerable manual operations, tend to be large as compared to the equivalent values at the corresponding station. High $c\theta$s (and since cs is typically less than 1 for manufacturing systems) imply $\gamma_i \geqq 0$. From Equation (3.6) we see that the relationship between the cas and αs is complex. However, Bitran and Tirupati [1989a] have shown that, for networks with large number of products, the cas are insensitive to capacity changes (see sections 4 and 5 for more discussion). Thus, for networks with large number of products, the impact of capacity variations in (3.6) can be ignored and system (3.6) can be approximated by

$$\lambda_i ca_i - \sum_j \lambda_j r_{ji}^2 (1-\rho_j^2) ca_j = \lambda_{0i} ca_i^0 + \gamma_i + \sum_j \lambda_j r_{ji} \left\{ r_{ji} \rho_j^2 cs_j + 1 - r_{ji} \right\}. \tag{3.9}$$

$$i = 1, 2...J$$

Equation (3.9) allows us to better understand the impact of the α_j's on the cas. From (3.9), the cas are linear functions of the γ_is. The γ_is are related to αs as in (3.8) and each α_i is a function of the σ_i^2. We now study the relationship between the γ_is and the σ^2s. From (3.8) we see that if the $c\theta$s are large and each α_j is convex in σ_j^2, then each γ_i is convex in the σ^2s. Since, from (3.9), the cas are linear in γs, δ^2s and σ^2s, they are convex in $\{\sigma^2, \delta^2\}$. The convexity of the cas implies that L_j for each workstation j is convex in $\{\sigma^2, \delta^2\}$ (the proof is the same as for the (VR) problem).

The WIP of each rework node j is given by the following expression.

$$\rho_j' + \frac{(\rho_j')^2}{2(1-\rho_j')} (cr_j + c\theta_j) g(\rho_j', cr_j, c\theta_j) \tag{3.10}$$

where

$\rho_j' = \alpha_j \lambda_j \theta_j$,

$cr_j = \alpha_j(1 - \rho_j^2)ca_j + \alpha_j \rho_j^2 cs_j + 1 - \alpha_j$, and

$g(\cdot)$ is defined in equation (2.1).

If the $c\theta$s are large, as we have assumed in the foregoing discussion, (3.10) suggests that rework WIPs are relatively insensitive to the changes in cr_j. To see why, note that the cr_j and $c\theta_j$ terms appear together, and crs are typically less than 1.0 for most manufacturing systems. One can, therefore, assume that the WIP of rework stations is mostly affected by the changes in the ρ_j' or equivalently, α_j. Since, by earlier discussion, each α_j is convex in σ_j^2, the WIP at each rework node j can be shown to be convex in σ_j^2. The result follows because (i) expression (3.10) is convex and increasing in ρ_j' (proven in Bitran and Tirupati [1989a]), and (ii) ρ_j' is convex and increasing in σ_j^2. Since the WIP at each workstation and the corresponding rework node is convex in $\{\sigma^2, \delta^2\}$ the methodology of the (VR) problem is applicable for the case when the routing of products is affected by variability reductions.

We conclude this section with a numerical example. The station 9 of the example in Appendix 3 has a rework node (station 14) with service time mean = 10.0 and scv = 2.0. Also, the initial α is 0.10. Now suppose that the service variability of station 9 is totally eliminated and this cuts the number of visits to the rework node by half to 0.05. The new total WIP is reduced by 21.8% to 25.96 (from 33.19).

4. Throughput Characterization Problem

In this problem our objective is to determine a schedule of capacity additions to achieve a target throughput. Specifically, we want to increase the throughput of all products by a factor $(1 + \beta^T)$. Initially, $\lambda_j = \lambda_j^I$, $\mu_j = \mu_j^I$ and total WIP $= W^I$. The throughput target is $(1 + \beta^T)\lambda_j^I$ for all j. Operationally, the throughput target is achieved by increasing the mean arrival rate of the external arrivals (λ_{oi}) by a factor $(1+\beta^T)$. The λ_{oi}s can be increased, for example, by increasing the batch sizes or introducing a new product. It is assumed that the λ_{oi}s can be increased without affecting the ca^o or cs at any node. At the end of this section we show that this assumption can be relaxed and the methodology

applies to other cases. In our first formulation we assume that the throughput target is to be achieved with minimum capacity additions cost and with WIP level below W^I. The throughput characterization problem can then be written as

$$(TC): \qquad \min \sum_{j=1}^{J} F_j(\mu_j) \tag{4.1a}$$

$$\text{s.t.} \quad (2.2),\ (2.3)$$

$$\sum_{j=1}^{J} v_j L_j \leqq W^I \tag{4.1b}$$

$$\mu_j \geqq \mu_j^I \quad j = 1,2,...,J \tag{4.1c}$$

$$\lambda_j \geqq (1 + \beta^T)\lambda_j^I \quad j = 1,2,...,J \tag{4.1d}$$

Some Modelling Assumptions

The cost functions $F_j(\mu_j)$s, as in Bitran and Tirupati [1989a], are assumed to be monotonically increasing, convex and differentiable in μ. The capacity increases in the model are achieved by increasing the μ_js. We assume that the mean and standard deviation of the service time vary in the same proportion, and hence, the css remain constant. When this is not the case, the results of this paper will apply as long as the variation in the css is such that the approximation to establish queue lengths (L_j) is convex in μ_j.

The functional relationship between capacity changes and the scvs of arrivals, cas is complex. The system of equations (2.3) is nonlinear and not easy to analyze. However, Bitran and Tirupati [1989a] have shown that in many situations, especially when the number of products is large (>8 or so), the sensitivity of the *cas* to capacity changes is small. Hence, we make the assumption that the scvs cas are invariant with capacity. (This assumption is relaxed in Section 5.) As a result of this assumption, systems (2.2) and (2.3) can be solved initially and the vectors cas and css can be treated as known parameters in (TC). Furthermore, using results from Bitran and Tirupati [1989a], (TC) reduces to a convex programming

problem. We propose a greedy heuristic for this resultant convex program as it readily provides the tradeoff curve.

In the heuristic we define a priority index (*PI*) for each station. *PI* is defined as the ratio of marginal benefit (improvement in throughput) to the marginal cost of capacity addition. At each iteration, we increase the capacity by an increment Δ at the station with the highest *PI*. The procedure is repeated until the throughput target is achieved. (The capacity increment Δ is chosen to balance accuracy needed and the computational requirements). At each iteration, we increase the capacity of the station with highest *PI* and compute the resultant throughput, $1 + \beta$, of the system by solving the following equation for β:

$$(1+\beta)\sum_{j=1}^{J}\frac{v_j\lambda_j^I}{\mu_j} + \frac{(1+\beta)^2}{2}\sum_{j:ca_j \geqq 1}\frac{v_j(\lambda_j^I)^2(ca_j+cs_j)}{\mu_j(\mu_j-(1+\beta)\lambda_j^I)}$$

$$+\frac{(1+\beta)^2}{2}\sum_{j:ca_j<1}\frac{v_j(\lambda_j^I)^2(ca_j+cs_j)}{\mu_j(\mu_j-(1+\beta)\lambda_j^I)}\, e^{-\frac{2}{3}\frac{(1-ca_j)}{(ca_j+cs_j)}\left[\frac{\mu_j}{(1+\beta)\lambda_j^I}-1\right]} = W^I \qquad (4.2)$$

If $\beta \geqq \beta^T$, we stop with a solution. The left hand side of (4.2) (henceforth referred to as $I(\beta, \mu)$) is monotone in β and thus, (4.2) can be solved for β using binary search or other efficient techniques. Furthermore, using (4.2) we compute the marginal improvement in throughput with respect to the capacity μ_j:

$$\frac{\partial\beta}{\partial\mu_j} = -\frac{\partial I}{\partial\mu_j} \Big/ \frac{\partial I}{\partial\beta} \qquad (4.3)$$

It is easy to show that $\frac{\partial\beta}{\partial\mu_j} > 0$ implying β is strictly monotone in the μs.

Greedy Heuristic for the Throughput Characterization Problem

Step 1: Initialization.

$\beta = 0$, $\lambda_j = \lambda_j^I$, $\mu_j = \mu_j^I$ and the corresponding $WIP = W^I$

Compute the priority indexed $PI_j = \frac{\partial\beta/\partial\mu_j}{\partial F_j/\partial\mu_j}$ at the current value of μ and β for each station j.

Step 2: If $\beta \geq \beta^T$, stop; else, go to Step 3.

Step 3: Let $j^* = argmax\ \{PI_j,\ j = 1,\ 2,\ \ldots J\}$ and set $\mu_{j^*} = \mu_{j^*} + \Delta$.

Update PI_j and β. The new β is the maximum value of β satisfying $I(\beta, \mu) \leq W^I$. Go to Step 2.

The heuristic converges in a finite number of steps. This can be seen as follows. For a conveniently small Δ, the reduction in WIP at any iteration due to capacity increment is greater than or equal to $\Delta.r_1$ where $r_1 = \min_i \frac{\partial L_i}{\partial \mu_i}$. The derivative is evaluated at $\lambda_i = \lambda_i^I$ and $\mu_i = \bar{\mu}_i$ where $\bar{\mu}_i$ is an upper bound on μ_i. Now, if β is increased by ε, at any iteration, the WIP increase (for small ε) is less than $r_2 \triangleq \varepsilon$. $\frac{\partial I}{\partial \beta}$ where the derivative is evaluated at $\beta = \beta^T$ and $\lambda_j = \lambda_j^I$ for all j. Since the WIP increment due to change in β and WIP decrement due to capacity increase must match in order for the WIP to equal W^I, we have $\varepsilon \geq r_1\ \Delta / r_2$. That is, at every iteration, β goes up by more than a finite amount implying finite convergence. Note that the heuristic produces an approximate tradeoff curve. Each iteration gives a point on the curve and by choosing Δ sufficiently small we can approximate the tradeoff curve reasonably well. In fact, using Kuhn-Tucker conditions, it is possible to show that the solution obtained from the heuristic is optimal in (TC) as $\Delta \rightarrow 0$. Since

(1) $I(\beta, \mu)$ is convex and increasing in β (the proof is straightforward and omitted),

(2) $I(\beta, \mu)$ is convex and decreasing in μ_j, and

(3) $F_j(\mu_j)$ is convex and increasing in μ_j,

the priority index PI_j is monotonically decreasing in μ_j for $\rho_j < 1$. Using this result and optimality conditions of problem (TC) we can show that $\mu_j^0 - \Delta \leq \mu_j^* \leq \mu_j^0$, where μ_j^* and μ_j^0 are the capacity of station j corresponding to the optimal and heuristic solutions respectively. Finally, using weak duality results, we can prove the following result that characterizes the performance of the heuristic.

Proposition: Let Z_0 and W^0 be the approximate values of the objective function and WIP obtained respectively by the heuristic. Also, let Z^* be the optimal value of the objective function. Then

$$0 \leqq \frac{Z_0 - Z^*}{Z^*} \leqq \frac{\varepsilon}{Z_0 - \varepsilon}$$

where

$$\varepsilon = u^0(W^I - W^0) + \Delta \sum_{j:\mu_j^0 > \mu_j^I} \left\{ F_j'(\mu_j^0) + u^0 v_j L_j'(\mu_j^0) \right\} \text{ with}$$

$$u^0 = \max_j \frac{\partial I(\beta^0, \mu^0)}{\partial \mu_j}$$

We conclude this section by commenting on alternate formulations and variations of the throughput characterization problem.

In formulation (TC) we put a limit on WIP costs and minimize the capacity investment costs. We could alternately formulate a problem in which WIP and investment costs are minimized simultaneously. We must, however, take care to model the one-time costs (such as training) and the variable costs (such as skilled labor, overtime) within the capacity addition costs. Let a' denote the amortization factor to distribute the capacity addition costs over time. Then the alternate formulation is written as

$$\text{minimize } S = a' \sum_{j=1}^{J} F_j(\mu_j) + \sum_{j=1}^{J} v_j L_j \tag{4.4}$$

subject to (2.2), (2.3), (4.1c) and (4.1d).

Again (4.4) is a convex programming problem (under the assumptions of (TC)). The earlier heuristic, with modified definition of $PI_j = \frac{\partial \beta / \partial \mu_j}{\partial S / \partial \mu_j}$, could be used to derive the throughput-capacity tradeoff curve. Since in practice it is quite common for managers to specify a WIP target level as it allows for better monitoring and improvement opportunities, the throughput characterization problem was described in terms of (3.1).

In the throughput characterization problem we assume that the throughput of all products (and hence, λs at all stations) are increased by the same factor. Our methodology, however, is easily extended to cases where the throughputs of different products are to be increased by different factors. This may arise, for example, when a new product or model of a product is introduced to the manufacturing system. In such situations different throughput factors allow to examine tradeoffs across product lines or product mix.

To roughly assess the throughput implications of capacity changes and process improvements, we again consider the example in Appendix 3. If the mean capacity or service rates of all stations are increased by 10%, the throughput of product 7, for example, can be increased by 138% without affecting the total WIP in the network. Alternatively, the 10% capacity improvement also translates to throughput increase of 11.2% for each of the 10 product types for the same total WIP in the network.

For the preceding analysis we assumed that the λ_js can be increased without affecting the ca^o or cs at a node. Whether this assumption holds would depend on the specific product release process to the job shop. The methodology above, however, applies to cases where this assumption need not hold. We conclude this section by pointing out some of these examples. Consider a queueing network in which (i) products arrive at stations in fixed size batches, (ii) jobs are processed individually at each station, and (iii) there is no service variability. Suppose that the throughput is increased by making the batch sizes bigger. Specifically, the batch size of product j is increased from b_j to $(1+\beta)b_j$. The WIP at station j is then approximated by

$$WIP_j = b_j(1+\beta)\, L_j((1+\beta)\lambda_j, \mu_j, ca_j) + \frac{b_j(1+\beta) + 1}{2}\, \rho_j$$

For this system the ca_i^2s are unaffected and accordingly the cas do not depend on β. We can also show that WIP_j is convex and increasing in β. This implies that the $I(\beta, \mu)$ [see equation (4.2)] is convex and increasing in β and the methodology applies. The throughput can also be increased by adding new products (without altering the batch sizes). As long as the changes in throughput make the cas increasing and convex in β it can be shown that the procedures developed in this section are directly applicable.

5. Throughput Characterization - Relaxation of Assumption

In solving the throughput characterization problem in Section 4, we, as in Bitran and Tirupati [1989a], made the assumption that the *cas* are insensitive to the capacity changes. This assumption, while reasonable in large networks, does not hold in many situations. In this section we relax it. For expositional simplicity, we consider the relaxation of this assumption in the WIP "targeting problem". The implications and analysis are identical for the throughput characterization problem. Algebraic manipulations for the targeting problem are somewhat simpler than the throughput characterization problem. The targeting problem (see Bitran and Tirupati [1989a] for detailed discussion) addresses the issue of capacity assignment to workstations in order to meet a target WIP-level while attaining minimal capacity costs. Specifically,

$$(TP): \qquad V = \min \sum_{j=1}^{J} F_j(\mu_j) \tag{5.1a}$$

s.t. (2.2), (2.3),

$$\sum_{j=1}^{J} v_j L_j \leqq W^T \tag{5.1b}$$

$$\mu_j \geqq \mu_j^l \quad j = 1,2,...,J \tag{5.1c}$$

Bitran and Tirupati conjectured that their greedy algorithm could be modified to solve for the case when the *cas* vary with capacity. In this section we propose an iterative scheme for the general case and establish sufficient conditions for its convergence.

The functional relationship between the μs and *cas* (system (2.3)) is complex. Consequently, it is not clear that (5.1) or (TP) is a convex programming problem. Thus, unless specified, in this section we search for a local solution to (TP). But, note that when the *cas* are given, (5.1) is a convex program. The iterative scheme uses this fact. Finally, the capacity-WIP tradeoff curve is constructed using the final values of the *cas* from the iterative algorithm.

Iterative Scheme to Solve (5.1)

Step 0: Initialization: choose initial ca_j^0 for all j.

Step 1: Solve

$$V(\underline{ca}) = \min \sum_{j=1}^{J} F_j(\mu_j) \qquad (5.2)$$

s.t. (5.1b) and (5.1c)

for current values of *cas*.

Any efficient techniques for Convex Programming Problems may be used here. Let $\underline{\mu}(ca)$ denote the optimal capacities corresponding to the given cas.

Step 2: Recompute the *cas* using (2.3) with the new values of capacities from step 1.

Stop, if the new and the last *cas* values are "close;" else, go to step 1.

We now discuss the convergence properties of this iterative scheme. First, note that the system of equations (2.3) can be abstractly represented as

$$\underline{ca} = \underline{G}(\mu_1, ..., \mu_j)$$

with $G: R^J \rightarrow R^J$. Then, the iterative scheme can be summarized as:

$$\underline{ca}^{k+1} = \underline{G}(\mu_1(\underline{ca}^k), \mu_2(\underline{ca}^k), ..., \mu_J(ca^k)) \qquad (5.3)$$

Using results from Ortega and Rheinboldt [1970], a sufficient (and essentially necessary) condition that the iterative scheme (5.3) converges is that the spectral radius of the Jacobian of the mapping G at the solution points is less than 1. That is,

$$\rho(G'(\underline{ca}^*)) < 1 \tag{5.4}$$

where $G'(\underline{ca}^*)$ is the Jacobian matrix of the mapping G at $\underline{ca}^*$ with elements G'_{ij}. Using (2.3), the expressions for G'_{ij}s are obtained as follows.

$$\lambda_i G'_{ii} = 2 \sum_{\substack{k=1 \\ k \neq i}}^{J} \frac{\lambda_k r_{ki}^2 \rho_k^2}{\mu_k} (ca_k - cs_k) \frac{\partial \mu_k}{\partial ca_i} \quad \text{and}$$

$$\lambda_i G'_{ij} = \lambda_j r_{ji}^2 (1 - \rho_j^2) + 2 \sum_{k \neq i} \frac{\lambda_k r_{ki}^2 \rho_k^2}{\mu_k} (ca_k - cs_k) \frac{\partial \mu_k}{\partial ca_j}, \quad i \neq j \tag{5.5}$$

While (5.4) and (5.5), evaluated at the solution points, together specify the most general sufficient condition, they require knowledge of μ^* and ca^* and hence, are not verifiable using the initial network data.

In what follows we derive apriori sufficient conditions that guarantee convergence of the iterative scheme. We restrict the value of ca_j between 0 and 1 since that reflects typical manufacturing situations. (A similar analysis can be performed when the cas are $\geqq 1$). We seek conditions that guarantee $\sum_j | G'_{ij} | < 1$, at the solution points, apriori ($\sum_j | G'_{ij} | < 1$ implies (5.4); see the discussion below). An upper bound on $| G'_{ij} |$ is established as follows. First, note that $\frac{\partial \mu_k}{\partial ca_j} \geqq 0$. This is because any increase in ca_j increases L_j and thus, the level of μ_k should either increase or stay the same in order to achieve the desired WIP target. Second, equation (5.1b) is used to characterize the relationship among the $\frac{\partial \mu_k}{\partial ca_j}$s. A change in ca_j implies changes in $\underline{\mu}(ca)$ of (5.2). Since (5.1b) holds, the changes can not be arbitrarily large. Specifically, differentiating (5.1b), we have

$$\sum_{k=1}^{J} v_k \left[\frac{\rho_k}{\mu_k} e^{\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{1-\rho_k}{\rho_k}} + \frac{\rho_k^2(ca_k+cs_k)(2-\rho_k)}{2\mu_k(1-\rho_k)^2} + \frac{(1-ca_k)\rho_k}{3\mu_k(1-\rho_k)} \right] e^{-\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{(1-\rho_k)}{\rho_k}} \cdot \frac{\partial \mu_k}{\partial ca_j}$$

$$= \left[\frac{v_j \rho_j^2}{2(1-\rho_j)} + \frac{v_j \rho_j (1+cs_j)}{3(ca_j+cs_j)} \right] e^{-\frac{2}{3}\frac{(1-ca_j)}{ca_j+cs_j}\cdot\frac{(1-\rho_j)}{\rho_j}} \tag{5.6}$$

We now provide a bound on the quantity

$$x_{ij} \triangleq \sum_k \frac{\lambda_k r_{ki}^2 \rho_k^2 \mid ca_k - cs_k \mid}{\mu_k} \cdot \frac{\partial \mu_k}{\partial ca_j}$$

which appears in equation (5.5). One such bound is obtained from the objective value of the following optimization problem:

$$\text{maximize} \sum_k \frac{\lambda_k r_{ki}^2 \rho_k^2 \mid ca_k - cs_k \mid}{\mu_k} \cdot \frac{\partial \mu_k}{\partial ca_j} \tag{5.7}$$

subject to (5.6)

Solution of (5.7) yields

$$x_{ij} \leqq \left\{ \left[\frac{\nu_j \rho_j^2}{2(1-\rho_j)} + \frac{\nu_j \rho_j (1+cs_j)}{3(ca_j+cs_j)} \right] e^{-\frac{2}{3}\frac{(1-ca_j)}{ca_j+cs_j}\frac{(1-\rho_j)}{\rho_j}} \right\}.$$

$$\max_k \frac{\lambda_r r_{ki}^2 \rho_k^2 \mid ca_k - cs_k \mid e^{\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{(1-\rho_k)}{\rho_k}}}{\nu_k \left[\rho_k e^{\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{(1-\rho_k)}{\rho_k}} + \frac{\rho_k^2 (ca_k+cs_k)(2-\rho_k)}{2(1-\rho_k)^2} + \frac{(1-ca_k)\rho_k}{3(1-\rho_k)} \right]} \tag{5.8}$$

From Ortega and Rheinboldt, if $\sum_j \mid G'_{ij} \mid < 1$ then $\rho(G') < 1$. From (5.5) and the definition of x_{ij}

$$\sum_j \mid G'_{ij} \mid \leqq \frac{2}{\lambda_i} \sum_j x_{ij} + \frac{1}{\lambda_i} \sum_j \lambda_j r_{ji}^2 (1-\rho_j^2).$$

So, if

$$\frac{2}{\lambda_i} \sum_{j=1}^{J} \mid x_{ij} \mid < 1 - \sum_{j=1}^{J} \frac{\lambda_j r_{ji}^2 (1-\rho_j^2)}{\lambda_i} \tag{5.9}$$

holds for all i at the solution point then $\sum_j \mid G'_{ij} \mid < 1$ and the iterative method converges. Combining (5.8) and (5.9), a sufficient condition for convergence is that, at solution points, the following must hold

$$\max_k \frac{\lambda_k r_{ki}^2 \rho_k \mid ca_k - cs_k \mid e^{\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{(1-\rho_k)}{\rho_k}}}{\nu_k \left[e^{\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{(1-\rho_k)}{\rho_k}} + \frac{\rho_k(ca_k+cs_k)(2-\rho_k)}{2(1-\rho_k)^2} + \frac{(1-ca_k)}{3(1-\rho_k)} \right]}$$

$$\leqq \frac{\lambda_i - \sum_{j=1}^{J} \lambda_j r_{ji}^2 (1-\rho_j^2)}{\sum_{j=1}^{J} \nu_j \rho_j \left\{ \frac{\rho_j}{2(1-\rho_j)} + \frac{(1+cs_j)}{3(ca_j+cs_j)} \right\} e^{-\frac{2}{3}\frac{(1-ca_j)}{ca_j+cs_j}\frac{(1-\rho_j)}{\rho_j}}}$$

$$i = 1, 2, \ldots, J \tag{5.10}$$

Examining (5.10) we see that a verifiable sufficient condition is

$$\max_k \frac{\lambda_k r_{ki}^2}{\nu_k} \max_{\substack{0 < \rho_k \leqq \rho_k^I \\ 0 \leqq ca_k \leqq 1}} \left\{ \frac{\rho_k \mid ca_k - cs_k \mid e^{\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{(1-\rho_k)}{\rho_k}}}{e^{\frac{2}{3}\frac{(1-ca_k)}{ca_k+cs_k}\frac{(1-\rho_k)}{\rho_k}} + \frac{\rho_k(ca_k+cs_k)(2-\rho_k)}{2(1-\rho_k)^2} + \frac{1-ca_k}{3(1-\rho_k)}} \right\}$$

$$\leqq \frac{\frac{1}{2}\left[\lambda_i - \sum_{j=1}^{J} \lambda_j r_{ji}^2\right]}{\sum_{j=1}^{J} \nu_j \rho_j^I \left\{ \frac{\rho_j^I}{2(1-\rho_j^I)} + \frac{(1+cs_j)}{3(1+cs_j)} \right\} e^{-\frac{2}{3}\frac{(1-\rho_j^I)}{cs_j\,\rho_j^I}}}$$

$$i = 1, 2, \ldots J \tag{5.11}$$

Since (5.11) implies (5.10), this is a sufficient condition. In deriving (5.11), we used the following facts

(i) $\rho_j^* \leqq \rho_j^I$,

(ii) the denominator in the right hand side of (5.10) is monotone increasing in ρ's, and

(iii) the left hand side of (5.11) is an upper bound on the left hand side of (5.10).

To recapitulate, if (5.11) holds then the iterative scheme to solve (TP) described earlier converges. The right hand side of (5.11) is known from the initial network data. Computing the left hand side of (5.11) requires solution to the two-variable inner optimization problem which can be obtained through numerical enumeration or other efficient techniques.

6. Conclusions

In this paper we have considered manufacturing systems modeled by open networks of queues and examined the tradeoffs between WIP, throughput, capacity and process improvements through variability reductions. We presented the throughput characterization (TC) and variability reduction (VR) problems. The (TC), under some mild assumptions, and (VR) problems were shown to be equivalent to convex programming problems. Greedy-type heuristics that facilitate the derivation of tradeoff curves were presented and analyzed for both problems. Finally, the implications of the assumptions made in developing the greedy procedure for the (TC) problem were also examined.

REFERENCES

[1] ALBIN, S. L. 1986. Delays for Customers From Different Arrival Streams to a Queue. *Mgmt. Sci.* 32, 329-340.

[2] AVRIEL, M. 1976. *Nonlinear Programming. Analyses and Methods.* Prentice-Hall, Englewood Cliffs, N.J.

[3] BITRAN, G. R., and D. TIRUPATI. 1988. Multiproduct Queueing Networks With Deterministic Routing: Decomposition Approach and the Notion of Interference. *Mgmt. Sci.* 34, 75-100.

[4] BITRAN, G. R. and D. TIRUPATI. 1989a. Tradeoff Curves, Targeting and Balancing in Manufacturing Queueing Networks. Opns. Res. 37, 547-564.

[5] BITRAN, G. R. and D. TIRUPATI. 1989b. Capacity Planning in Manufacturing Networks with Discrete Options. Annals of Opns. Res. 17, 119-136.

[6] BOXMA, O. J., RINNOOY KAN, A.H.G. and Van VLIET, M. 1990. Machine Allocation Problems in Manufacturing Networks. E.J.O.R. 45, 47-54.

[7] BUZACOTT, J. A., and J. G. SHANTHIKUMAR. 1985. Approximate Queueing Models of Dynamic Job Shops. *Mgmt. Sci.* 31, 870-887.

[8] HALL, R. 1983. *Zero Inventories*, Dow Jones-Irwin, Homewood, Ill.

[9] HAYES, R. H., and WHEELWRIGHT. 1984. *Restoring Our Competitive Edge. Competing Through Manufacturing.* John Wiley & Sons, New York.

[10] IMAI, M. 1986. KAIZEN, *The Key to Japan's Competitive Success.* Random House, NY.

[11] KRAEMER. W., and M. LANGENBACH-BELZ. 1976. Approximate Formulae for the Delay in the Queueing System GI/G/1. Congressbook, 8th ITC, Melbourne. 235.1-235.8.

[12] ORTEGA, J. M. and RHEINBOLDT, W. C. 1970. *Iterative Solutions of Nonlinear Equations in Several Variables.* Academic Press, NY.

[13] SARKAR, D. and W. I. ZANGWILL. 1988. Variance Effects in Cyclic Production Systems. To appear in *Mgt. Sci.* (April, 1991).

[14] SCHONBERGER, R. 1982. *Japanese Manufacturing Techniques.* The Free Press, NY.

[15] SEGAL, M. and W. WHITT, "A Queueing Network Analyzer for Manufacturing," Proc. 12th Internat. Teletraffic Congress, Torino, Italy, June 1988.

[16] SHANTHIKUMAR, J. G. and J. A. BUZACOTT. 1981. Open Queueing Network Models of Dynamic Job Shops. *Int. J. Prod. Res.* **19**, 255-266.

[17] SKINNER, W. 1974. The Focussed Factory. *Harvard Bus. Rev.* (May-June), pp. 113-121.

[18] WHITT, W. 1983a. The Queueing Network Analyzer. *Bell Syst. Tech. J.* **62**, 2779-2815.

[19] WHITT, W. 1983b. Performance of the Queueing Network Analyzer. *Bell Syst. Tech. J.* **62**, 2817-2843.

[20] WHITT, W. 1985. Approximates for the GI/G/m Queue. *Adv. Appl. Prob.* (to appear).

Appendix 1

Convexity of L_j

To prove that L_j is jointly convex in ca_j and cs_j we assume that $ca_j \leqq 1$ (the result is obvious when $ca_j > 1$). We have that

$$L_j = \rho_j + \frac{\rho_j^2 (ca_j + cs_j)}{2(1 - \rho_j)} \cdot e^{-\frac{2}{3}\frac{(1-ca_j)}{ca_j + cs_j}\frac{(1-\rho_j)}{\rho_j}}$$

For notational simplicity we will prove the convexity of L in x and y where

$$L = \alpha(x+y)\, e^{-\beta\frac{(1-x)}{x+y}}$$

Denoting the derivatives of L with respect to x and y by L_x and L_y respectively, we have

$$L_x = \left[\alpha + \alpha(x+y)\left\{-\beta\frac{-(x+y) - (1-x)}{(x+y)^2}\right\}\right] e^{-\frac{\beta(1-x)}{x+y}} = \left[\alpha + \frac{\alpha\beta(1+y)}{x+y}\right] e^{-\frac{\beta(1-x)}{x+y}}$$

$$L_y = \left[\alpha + \alpha(x+y)\left\{-\beta(1-x)\cdot\left[-\frac{1}{(x+y)^2}\right]\right\}\right] e^{-\frac{\beta(1-x)}{x+y}} = \left[\alpha + \frac{\alpha\beta(1-x)}{x+y}\right] e^{-\frac{\beta(1-x)}{x+y}}$$

Also,

$$L_{xx} = \frac{\alpha\beta^2(1+y)^2}{(x+y)^3} e^{-\frac{\beta(1-x)}{x+y}}$$

$$L_{yy} = \frac{\alpha\beta^2(1-x)^2}{(x+y)^3} e^{-\frac{\beta(1-x)}{x+y}}$$

$$L_{xy} = \frac{\alpha\beta^2(1-x)(1+y)}{(x+y)^3} e^{-\frac{\beta(1-x)}{x+y}}$$

Hence,

$$A = \begin{bmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{bmatrix} = \frac{\alpha\beta^2}{(x+y)^3} e^{-\frac{\beta(1-x)}{x+y}} \begin{bmatrix} (1+y)^2 & (1+y)(1-x) \\ (1+y)(1-x) & (1-x)^2 \end{bmatrix}$$

Since matrix A is positive semi-definite, L is jointly convex in x and y.

Appendix 2

Derivation of PI′ and PI″

Step 1: Compute the inverse of the following matrix:

$$\begin{bmatrix} \lambda_1 - \lambda_1 r_{11}^2(1-\rho_1^2) & -\lambda_2 r_{21}^2(1-\rho_2^2) & -\lambda_3 r_{31}^2(1-\rho_3^2) & \cdots \\ -\lambda_1 r_{12}^2(1-\rho_1^2) & \lambda_2 - \lambda_2 r_{22}^2(1-\rho_2^2) & -\lambda_3 r_{32}^2(1-\rho_3^2) & \cdots \\ \cdots & \cdots & & \\ \cdots & \cdots & & \\ \cdots & \cdots & & \end{bmatrix}$$

Denote the inverse of this matrix by $A \triangleq ((a_{ij}))$. Note that the above matrix has Leontief structure and many efficient techniques are available to compute A.

Step 2: Compute (i) *cas* at the current value of δ^2 and σ^2 using (4.1), and (ii) $\frac{\partial L_j}{\partial ca_j}$, $\frac{\partial L_j}{\partial cs_j}$.

Step 3: Compute $\frac{\partial ca_i}{\partial \delta_k^2}$ for $i, = 1, 2, \ldots J$ and $k \in E$.

Note that

$$\lambda_i \frac{\partial ca_i}{\partial \delta_k^2} - \sum_{j=1}^{J} \lambda_j r_{ji}^2 (1 - \rho_j^2) \frac{\partial ca_j}{\partial \delta_k^2} = \begin{cases} 0 & \text{if} \quad k \neq i \\ \lambda_{0k}^3 & \text{if} \quad k = i \end{cases}$$

and thus

$$\frac{\partial ca_i}{\partial \delta_k^2} = a_{ik} \cdot \lambda_{0k}^3$$

Step 4: Compute $\frac{\partial ca_i}{\partial \sigma_k^2}$ for $i, k = 1, 2 \ldots J$

From (4.1c)

$$\lambda_i \frac{\partial ca_i}{\partial \sigma_k^2} - \sum_j \lambda_j r_{ji}^2 (1 - \rho_j^2) \frac{\partial ca_j}{\partial \sigma_k^2} = \lambda_k^3 r_{ki}^2$$

and consequently,

$$\frac{\partial ca_i}{\partial \sigma_k^2} = \sum_{j=1}^{J} a_{ij} \lambda_k^3 r_{kj}^2 \quad \text{for } i, k = 1, 2, \ldots J$$

Step 5: Compute PI' and PI''.

Since

$$\frac{\partial W}{\partial \delta_k^2} = \sum_{j=1}^{J} v_j \frac{\partial L_j}{\partial \delta_k^2} = \sum_{j=1}^{J} v_j \frac{\partial L_j}{\partial ca_j} \cdot \frac{\partial ca_j}{\partial \delta_k^2}$$

$$\frac{\partial W}{\partial \sigma_k^2} = \sum_j v_j \frac{\partial L_j}{\partial \sigma_k^2} = \sum_j v_j \left\{ \frac{\partial L_j}{\partial ca_j} \frac{\partial ca_j}{\partial \sigma_k^2} + \frac{\partial L_j}{\partial cs_j} \frac{\partial cs_j}{\partial \sigma_k^2} \right\}$$

$$\frac{\partial cs_j}{\partial \sigma_k^2} = \begin{cases} 0 & \text{if } j \neq k \\ \mu_k^2 & \text{if } j = k \end{cases}.$$

and all other quantities in the right hand side are known (Steps 2-4) above), $\frac{\partial W}{\partial \delta_k^2}$, $\frac{\partial W}{\partial \sigma_k^2}$ are known. Finally, PI' and PI'' are computed using (4.4) and (4.5).

Appendix 3

Data for the Network Example

Number of stations = 14
Number of products = 10

TABLE 1
Product Routing Characteristics

Product	Number of Operations	Routing Sequence*	
1	7	1, 2, 4, 2, 9, 10, 11	with prob 0.9
	8	1, 2, 4, 2, 9, 14, 10, 11	with prob 0.1
2	8	1, 2, 5, 2, 8, 9, 10, 11	with prob 0.9
	9	1, 2, 5, 2, 8, 9, 14, 10, 11	with prob 0.1
3	8	1, 2, 6, 4, 2, 9, 12, 11	with prob 0.9
	9	1, 2, 6, 4, 2, 9, 14, 12, 11	with prob 0.1
4	8	1, 2, 7, 4, 2, 9, 10, 11	with prob 0.9
	9	1, 2, 7, 4, 2, 9, 14, 10, 11	with prob 0.1
5	8	1, 2, 4, 12, 2, 9, 2, 13	with prob 0.9
	9	1, 2, 4, 12, 2, 9, 14, 2, 13	with prob 0.1
6	8	1, 2, 5, 12, 2, 9, 7, 13	with prob 0.9
	9	1, 2, 5, 12, 2, 9, 14, 7, 13	with prob 0.1
7	8	1, 2, 6, 12, 2, 8, 2, 13	
8	12	1, 2, 3, 7, 4, 12, 2, 8, 6, 9, 2, 13	with prob 0.9
	13	1, 2, 3, 7, 4, 12, 2, 8, 6, 9, 14, 2, 13	with prob 0.1
9	13	1, 2, 3, 5, 4, 6, 12, 2, 8, 2, 10, 6, 13	
10	13	1, 2, 3, 6, 2, 4, 12, 7, 2, 9, 11, 5, 13	with prob 0.9
	14	1, 2, 3, 6, 4, 12, 7, 2, 9, 14, 11, 5, 13	with prob 0.1

* The routing sequence for each product specifies the stations visited by the product in order.

TABLE 2

Interarrival Times

(Note: The mean interarrival time is 10.0 for jobs in each product class in all cases.)

Product	scv
1	0.333
2	0.500
3	0.333
4	0.333
5	0.250
6	0.500
7	0.250
8	0.333
9	0.250
10	0.500

TABLE 3

Service Times

Station	Mean	scv
1	0.78	0.333
2	0.348	0.333
3	2.67	0.5
4	1.05	1.0
5	2.0	0.333
6	1.4	0.25
7	1.775	0.333
8	1.875	0.333
9	1.175	0.5
10	1.8	0.333
11	1.44	1.0
12	1.158	0.25
13	1.45	0.333
14	10.00	2.0

TABLE 4

Utilization of Stations under $\lambda_i = 0.1$

Station	Utilization
1	0.78
2	0.87
3	0.80
4	0.74
5	0.80
6	0.84
7	0.71
8	0.75
9	0.94
10	0.72
11	0.72
12	0.81
13	0.87
14	0.80