〈硏究論文〉

Journal of the Korean Institute of Industrial Engineers Vol. 15, No. 1, June. 1989

The Workload Distribution Problems in a Class of Flexible Manufacturing Systems

Kim, Sung Chul*

Abstract

This study complements the previous studies on workload distribution problems in Flexible Manufacturing Systems. Specifically, we consider the problem in two perspectives, the long-range policy and the short and medium-term planning and control. The long-term loading policy focusses on identifying the optimal loading of the system characterized by either balanced loading or unique unbalanced loading for which a steepest ascent method is developed. These results are then applied to study the optimal medium and short-term planning and control problems, for which a truncated dynamic programming method is developed in order to obtain the optimal allocation of the given operation mix of part types to work stations.

1. Introduction

The growing interest in the development and implementation of Flexible Manufacturing Systems(FMSs) brought many new problems which changed our concept of management of manufacturing systems. One such problem in FMSs is the workload distribution problem, usually termed as 'loading problem', which can give an insight into production planning and control of FMSs.

A common practice in non-automated jobshops and flow shops is to assign each operation to one machine type. In case of FMS, the individual machine with automatic tool interchange capability can perform many different types of operations. These versatile machines with automated material handling devices allow to route jobs automatically thr-

* Department of Business, Administration, Duk Sung Women's University

ough the manufacturing system from one machine to the next under the computer control and numerical control techniques. And the flexibility in assigning operations among machines permits very general flow patterns of the jobs within the system. Therefore, as an automated alternative to the traditional means of batch manufacturing, managing production for an FMS requires more complex planning and control for part routing, tool interchange, and machine operations.

Typically, the loading problem arises mainly due to the limited tool magazine capacity. To route a part to a particular station, all the tools required for the operation should be placed within its limited capacity tool magazine. Therefore, the loading problem basically concerns how to allocate all the operations of the product mix of part types among machines within the associated technical constraints ; in other words, how to distribute the work requirement of part types among machines to determine which machine will be able to perform each operation of part types.

This paper places a special emphasis on two types of loading problems : one with respect to the long-term policy and the other with respect to the short-medium-term operational planning. The objective of the first one is to predict the performance of the system in the long-term perspective not being sensitive to fluctuations in the part types and operations mix which directly affect short-term production planning and part scheduling. The second problem arises on the operational level. That is, how to allocate given operation mix to stations within the long-term policy already predicted and its associated technical constraints. Each operation must therefore be assigned to stations, which is particularly of interest to managers in short and medium term planning.

In section 2, we define the loading problem on the long-term perspective and formulate it. In section 3, we develop a search method to obtain the optimal loading policy. Section 4 applies the result of section 3 to solve the operational planning problem described. A numerical example is presented in section 5 to illustrate the potential of the model. Finally, section 6 concludes the paper with brief summary.

2. Problem Formulation

Suppose we have product types to be manufactured and their relative product ratio, then we have the number of each operation types from the operation mix of part types. Let J be the number of part types and M be the number of stations in the shop. Also, let ω_j be the number of type j operations, j=1, ⋯, J, υ_i be the number of operations assigned to station i, i=1, ⋯, M, and S_j be the processing time of the type j operation, j=1, ⋯, J

Define a variable χ_{ij} which represents the number of operation type j assigned to station i, i=1, ⋯, M and j=1, ⋯, J. Then, the number of type j operations, ω_j, and the number of operations assigned to station i, υ_i, can be written respectively as :

$$\omega_j = \sum_{i=1}^{M} \chi_{ij} \qquad (j=1, \cdots, J), \tag{2.1}$$

$$\upsilon_i = \sum_{j=1}^{J} \chi_{ij} \qquad (i=1, \cdots, M). \tag{2.2}$$

Let μ_i be the service rate of machine at station i, then the mean processing time of machine at station i, denoted by $1/\mu_i$, simply takes the weighted average of the processing times of the operations performed at station i. That is:

$$1/\mu_i=\sum_{j=1}^{J} S_j\chi_{ij}/\sum_{j=1}^{J}\chi_{ij} \qquad (i=1, \cdots, M). \tag{2.3}$$

The average work requirement assigned to station i, denoted by ρ_i, is equal to the product of the mean processing time of a machine at station i and the number of operations assigned to station i, which is equivalent to

$$\rho_i=\upsilon_i(1/\mu_i)=\sum_{j=1}^{J} S_j\chi_{ij} \qquad (i=1, \cdots, M), \tag{2.4}$$

and can be used as a measure of the relative workload assigned to station i. Equivalently, $\sum\rho_i$, where

$$\sum_{i=1}^{M}\rho_i=\sum_{i=1}^{M}\sum_{j=1}^{J} S_j\chi_{ij}=\sum_{j=1}^{J} S_j\omega_j \tag{2.5}$$

is the average work requirement of jobs within the system. Then what we are concerned with is how to distribute the average work requirement of the system among stations.

A major estimation of performance can be measured through the evaluation of the Jackson queueing network(Jackson 1963) which has been extensively applied to study Flexible manufacturing Systems(Buzacott & Yao 1986). However, the analysis is mainly on the closed queueing network(Gorden & Newell 1967) due to the fact that the results for the closed queueing network can be directly extended to the other Jackson networks and that efficient algorithms are also available to drive both performance measures and queue length distributions(Bruell & Balbo 1980, Reiser 1981).

Consider a closed queueing network with M stations and N jobs which are determined by the number of pallets within the system. Station i has s_i parallel servers. Each server at station i has a service rate $\mu_i(0<\mu_i<\infty)$, i=1, ⋯, M, and the number of operations υ_i, i=1, ⋯, M, assigned. Let $\rho_i=\upsilon_i/\mu_i$, i=1, ⋯, M, be the average work requirement assigned to station i, which is commonly denoted as service intensity at station i. Denote n_i be the total number of jobs at station i(including both jobs in queue and jobs in service).

Then the throughput function of the closed queueing network can be described as:

$$TH(N)=G(N-1)/G(N), \tag{2.6}$$

where G(N) is the normalizing constant of the closed queueing network with N jobs:

$$G(N)=\sum_{\sum n_i=N}\prod_{i=1}^{M}\rho_i^{n_i}/\theta_i(n_i) \tag{2.7}$$

$$\theta_i(n_i)=\begin{cases} n_i!, & 0\leqq n_i\leqq s_i, \\ s_i!\, s_i^{n_i-s_i}, & s_i\leqq n_i\leqq N, \end{cases} \tag{2.8}$$

and

$$\rho_i=\sum_{j=1}^{M} S_j\chi_{ij}. \tag{2.9}$$

Let $\underline{\chi}=(\chi_{11}, \chi_{12}, \cdots, \chi_{MJ})$, then we are basically concerned with studying $TH(\underline{\chi})$ as a function of $\underline{\chi}$. Also notice that TH(N) and $TH(\underline{\chi})$ will be used interchangably in the sequel for the notational convenience. Therefore the problem at hand may now be formulated as:

(P 1)

Maximize $TH(\underline{\chi})$ (2.10)

s.t. $\sum_{j=1}^{M} \chi_{ij}=\omega_j \quad (j=1, \cdots, J),$ (2.11)

$\chi_{ij} \geq 0$ and integer $\quad (i=1, \cdots, M \ \& \ j=1, \cdots, J).$ (2.12)

That is then to find a partition of the given a set of integers, e.g. $\omega_1, \omega_2, \cdots, \omega_J$, such that the throughput of the closed queueing network is maximized. Then the number of the distinguished states of the distribution of the average work requirement of the system is equal to the number of partitions of$(\omega_1, \cdots, \omega_J)$ operations among M stations, $\prod_{j=1}^{M} \binom{M+\omega_j-1}{M-1}/M!$, which is equivalent to the number of evaluations of the throughput function by the total enumeration of the states.

However, there are other constraints to be considered with respect to the tool magaz ine capacity. That is, given the total number of slots of a tool magazine, the number of operation types that can be assigned to station i is bounded. Suppose T is the number of slots of the tool magazine and t_j is the number of slots required by operation type j. Also there can be some tool slot savings by assigning operations which share the same tools to the same station.

Since the decision variable χ_{ij}, $i=1, \cdots, M$ & $j=1, \cdots, J$, is not restricted to be 0 and 1, to formulate properly the tool magazine capacity constraints, we need to redefine each χ_{ij} as a linear combination of $0-1$ variables,

$$\chi_{ij}=\sum_{k=1}^{\omega_j} k y_{ijk} \quad (i=1, \cdots, M \ \& \ j=1, \cdots, J), \tag{2.13}$$

with constraints

$$\sum_{k=1}^{\omega_j} y_{ijk} \leq 1 \quad (i=1, \cdots, M \ \& \ j=1, \cdots, J) \tag{2.14}$$

and $y_{ijk}=(0, 1) \quad (i=1, \cdots, M, j=1, \cdots, J, k=1, \cdots, \omega_j).$ (2.15)

Therefore if $\sum_{k=1}^{\omega_j} y_{ijk}=1$, the operation type j is assigned to station i and t_j slots are required.

Denote $\Lambda=\{1, \cdots, J\}$ to be the set of all operation types and s be the subset of Λ. Let t_s be the number of slots shared by all the operations in subset s. Also denote $|s|$ be the cardinality of subset s. Then the tool magazine capacity constraints can be formulated as :

$$\sum_{j=1}^{J} t_j\left(\sum_{k=1}^{\omega_j} y_{ijk}\right)+\sum_{|s|=2}^{J}(-1)^{|s|+1}\sum_s t_s \Pi_{j\in s}\left(\sum_{k=1}^{\omega_j} y_{ijk}\right) \leq T, \quad (i=1, \cdots, M). \tag{2.16}$$

Therefore, the problem can be reformulated as :

(P 2)

Maximize $TH(\underline{Y})$

s.t. $\sum_{i=1}^{M} \sum_{k=1}^{\omega_j} k y_{ijk}=\omega_j \quad (j=1, \cdots, J),$ (2.17)

(2.14),

(2.15),

(2.16),

where $\underline{Y}=(y_{111}, y_{112}, \cdots, y_{MJ\omega_J})$.

3. The Optimal Loading Policy

The different loading policies have the different average work requirements of jobs among stations. Suppose the average work requirements of jobs within the system is fixed, that is $\rho_1+\cdots+\rho_M=L$, then a given loading policy is represented by a loading vector $\underline{\rho}=(\rho_1, \cdots, \rho_M)$.

A known conclusion in the literature is that the balanced loading in optimal is FMSs with single machine stations(Buzacott & Shanthikumar 1980, Shanthikumar 1982, Shanthikumar & Stecke 1986, Yao 1985, 1987). Specifically, for all loading vectors $\underline{\rho}$ that satisfy $|\underline{\rho}|=\rho_1+\cdots+\rho_M=L$, where L is given constant, the balanced loading $\underline{\rho}^*=(L/M, \cdots, L/M)$ maximizes the system throughput. It is proved that the optimality of balanced loading also holds for FMSs with each station having the same number of multiple parallel machines (Yao & Kim 1987 a, b). However, in case of FMSs with stations of multiple parallel machines of unequal sizes, the balanced loading is not optimal(Stecke 1985), and the way to find optimal loading has not yet studied.

Therefore, in this section we concentrate on FMSs which have a different number of machines among stations and also on finding a method which can obtain the optimal loading solution.

The solution method we propose is an iterative procedure, finding a search direction, and. if the gradient of the throughput, TH(N), with respect to ρ_i, i=1, $\cdots$, M, can be found, the steepest decent method can be identified. From (2.7), we have

$$G(N)=\sum_{\sum n_i=N}\prod_{i=1}^{M} \rho_i^{n_i}/\theta_i(n_i)$$
$$=\sum_{n_i=0}^{N}(\rho_i^{n_i}/\theta_i(n_i))\, G_i(N-n_i) \qquad (3.1)$$

where $G_i(N-n_i)$ is normalization constant of a closed queueing network with $N-n_i$ jobs and $M-1$ stations where station i is excluded.

$$\partial G(N)/\partial\rho_i=\sum_{n_i=1}^{N} n_i(\rho_i^{n_i-1}/\theta_i(n_i))G_i(N-n_i)$$
$$=(1/\rho_i)\sum_{n_i=1}^{N} n_i(\rho_i^{n_i}/\theta_i(n_i))G_i(N-n_i). \qquad (3.2)$$

Since $(1/G(N))\sum_{n_i=1}^{N} n_i(\rho_i^{n_i}/\theta_i(n_i))\ G_i(N-n_i)$ is simply the mean queue length of station i, we have

$$\partial G(N)/\partial\rho_i=(G(N)/\rho_i)L_i(N) \qquad (3.3)$$

where $L_i(N)$ is the mean queue length of station i including jobs in service when there are n jobs in the system. Using this result, the following can be shown :

$$\partial TH(N)/\partial\rho_i=(\partial/\partial\rho_i)(G(n-1)/G(N))$$
$$=(1/\rho_i G(N))/G(N-1)\{L_i(N-1)-L_i(N)\}$$
$$=-(TH(N)/\rho_i)\{L_i(N)-L_i(N-1)\}. \qquad (3.4)$$

In view of the deviation procedure, it is an extention of Kobayashi & Gerla(1983)'s result in computer networks with single server stations to the FMSs with multiple parallel machine stations.

The negative gradient in (3.4) is termed as the direction of the steepest descent. Therefore, at iteration k, to increase the throughput, compute gradient(i)=$(1/\rho_i)\{L_i(N)-L_i(N-1)\}$ for i=1, ⋯, M, and modify $\rho_i^{(k+1)}=\rho_i^{(k)}-\triangle$ with max gradient(i) and $\rho_i^{(k+1)}=\rho_i^{(k)}+\triangle$ with min gradient(i) and $\rho_i^{(k+1)}=\rho_i^{(k)}$ for the other stations. As an initial loading, e.g. $\underline{\rho}^{(0)}$, empirical study suggests that when N is large, the balanced loading is better and when N is small, the stations which have more servers should have larger ρ_i's. However, there is no clear cut criteria, since as the number of jobs in the system decreases, the degree of unbalancing of the optimal loading increases.

An algorithm to find optimal loading can be summarized as follows :

0. Set k=0 ; let $\underline{\rho}^{(0)}$ be the initial loading and compute $TH^{(0)}(N)$.
1. Compute gradient(i) for all i,
   and set $\rho_\gamma^{(k+1)}=\rho_\gamma^{(k)}-\triangle$ and γ=max arg{gradient(i), $i \in M$}
   $\rho_\alpha^{(k+1)}=\rho_\alpha^{(k)}+\triangle$ and α=min arg{gradient(i), $i \in M$}
   $\rho_i^{(k+1)}=\rho_i^{(k)}$ for all other $i \neq \gamma, \alpha$.
2. Compute $TH^{(k+1)}(N)$.
   If $TH^{(k+1)}(N)-TH^{(k)}(N)<\varepsilon$, where ε is a properly chosen limit, stop. Else go to 3.
3. Let k=k+1 and go to 1.

Step 1 considers means of characterizing the set of feasible points in a neighborhood of feasible points $\underline{\rho}^{(k)}$, given the fixed average work requirement of jobs within the system. To increase the convergence of the algorithm, some other nonlinear techniques can also be adopted(Gill, Murray & Wright 1981). Also, $\triangle$ denotes step length which should be chosen deliberately to increase the convergence. However, to be properly used, the convexity of the throughput function is required. Though all the numerical results show that the above algorithm leads to the global optimal solution, the convexity of the throughtput function can not be established, and we leave it as a conjecture.

To summarize, when the number of machines is the same for all stations, the optimal loading can easily be identified, the result of which is termed as balanced loading. When the number of machines is different among stations, the optimal loading can be determined by the iterative method described above, which can be termed as a unique unbalanced loading.

4. The Loading Problem in Operational Level

In the following discussion, we concentrate on the solution procedure of problem (P 2) in section 2. To facilitate the solution procedure, an alternative loading objective can be defined based on the result of section 3. The rationale for the new objective function is that we want the deviation of the actual loading from the optimal loading to be small. One way to accomplish this is to minmize the sum of squares of the deviations. Thus, if ρ_i is the actual loading of station i, then the deviation of ρ_i from the optimal loading ρ_i^* is $\rho_i-\rho_i^*$,

$i=1, \cdots, M$, and the sum of squares of deviations to be minimized is:

$$\sum_{i=1}^{M} (\rho_i - \rho_i^*)^2 = \sum_{i=1}^{M} \left(\sum_{j=1}^{J} S_j \sum_{k=1}^{\omega_j} k y_{ijk} - \rho_i^*\right)^2. \quad (4.1)$$

Therefore, (P 2) can be modified as:

(P 3)

$$\begin{aligned} \text{Min.} \quad & \sum_{i=1}^{M}\left(\sum_{j=1}^{J} S_j \sum_{k=1}^{\omega_j} k y_{ijk} - \rho_i^*\right)^2 \\ \text{s.t.} \quad & (2.17), \\ & (2.14), \\ & (2.15), \\ & (2.16). \end{aligned}$$

The first approach to solve (P 3) is direct application of available nonlinear mixed integer programming techniques. Also, by utilizing the fact that all the nonlinear terms of (P 3) are products of $0-1$ integer variables, the methods of linearization of the product terms can be used. For further discussions of the solution methods of these approaches, we refer to Stecke(1983) and references therein.

The second method is rather based on the original variable χ_{ij}, $i=1, \cdots, M$, $j=1, \cdots, J$, of formulation (P 1) and a dynamic programming formulation is developed to obtain the optimal distribution of operations mix to stations. Combined with an approach similar to Lawler and Bell's method(1967), the optimal allocation of operations mix can be obtained with the considerably smaller than the total number of possible allocations. The number of operation types to be distributed is getting bigger, the degree of reduction of computation increases, and the total number of computation is bounded by the total number of slots, T, and the optimal workload, ρ_i^*, $i=1, \cdots, M$, in a sense.

To explain, let $\underline{\chi}_i=(\chi_{ij})_{j=1}^{J}$, $i=1, \cdots, M$, and $\underline{\chi}_j=(\chi_{ij})_{i=1}^{M}$, $j=1, \cdots, J$, be the vectors of decision variables and $\Phi_i(\underline{\chi}_i)$ be the objective function given by$(\sum_{j=1}^{J} S_j \sum_{k=1}^{\omega_j} k y_{ijk} - \rho_i^*)^2$ for all station i, $i=1, \cdots, M$. The objective function to be minimized is $\Omega(\underline{\chi}_1, \cdots, \underline{\chi}_J)=\sum_{i=1}^{M} \Phi_i(\chi_{i1}, \cdots, \chi_{iJ})$. Let $\Omega_k(A_{k1}, \cdots, A_{kJ})=\min \sum_{i=1}^{k} \Phi_i(\chi_{i1}, \cdots, \chi_{iJ})$ where the minimum is taken over all χ_{i1}'s, $\cdots$, χ_{iJ}'s such that $\sum_{i=1}^{k} \chi_{ij}=A_{kj}$, $j=1, \cdots, J$. Then from the principle of optimality, we have, for $i=2, \cdots, M$,

$$\Omega_k(A_{k1}, \cdots, A_{kJ}) = \min_{\substack{0 \leqq x_{kj} \leqq A_{kj} \\ j=1, \cdots, J}} \{(\Phi_k(\chi_{k1}, \cdots, \chi_{kJ}) + \Omega_{k-1}(A_{k1}-\chi_{k1}, \cdots, A_{kJ}-\chi_{iJ})\} \quad (4.2)$$

for all $A_{kj}=0, \cdots, \omega_j$, $j=1, \cdots, J$. The boundaries are:

$$\Omega_1(A_{11}, \cdots, A_{1J})=\Phi_1(A_{11}, \cdots, A_{1J}),\ 0 \leqq A_{1j} \leqq \omega_j,\ j=1, \cdots, J. \quad (4.3)$$

Then, it is easily verified that the number of operation of the dynamic programming is 0 $(M \prod_{j=1}^{J} \omega_j^2)$.

Let $\underline{\chi}_i=(\chi_{ij})_{j=1}^{J}$ and $\underline{y}_i=(y_{ij})_{j=1}^{J}$, then a vector partial ordering, $\underline{\chi}_i < \underline{y}_i$, is defined as $\chi_{ij} < y_{ij}$ for all $j \in \Lambda$. For example, two evctors (1, 3, 2, 1) and (2, 3, 3, 2) can be ordered in the vector partial ordering but (3, 2, 1, 1) and (2, 3, 1, 1) are noncomparable. It should be noted that the vectors following an arbitrary vector $\underline{\chi}_j$ be either greater than or noncomparable with $\underline{\chi}_j$ in the partial ordering. In the process of dynamic programming, at a certain stage i, the possible states of the allocation of the operation mix consists of a set of sequence of vectors which satisfies the vector partial ordering. For example, if the concept of array with M dimension is used, in a looping process, the most inner loop corresponds to operation type J and a sequence of vectors$(\chi_{i1}, \cdots, \chi_{iJ-1}, 0), \cdots, (\chi_{i1}, \cdots, \chi_{iJ-1}, \omega_J)$ is a set of vectors which can be ordered under the vector partial ordering.

Define another variable whose meaning is obvious by itself. Let $\delta_{ij}=1$ if $\chi_{ij}>0$, and $\delta_{ij}=0$, otherwise. If $\delta_{ij}=1$, then operation type j is assigned at station i and $\delta_{ij}=0$, otherwise. Then the constraint (2.16) of (P 3) can be modified to:

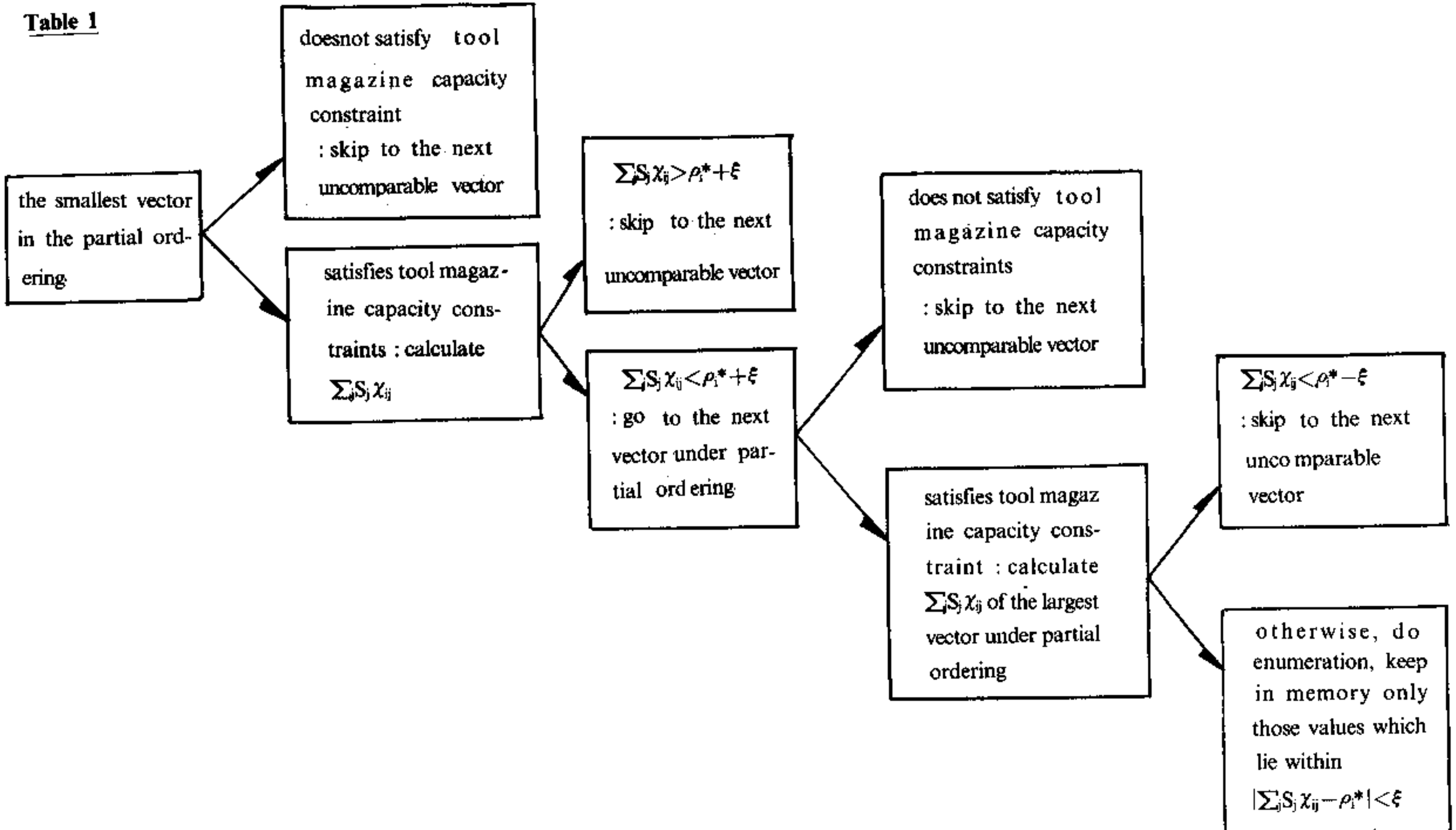
$$\sum_{j=1}^{J} t_j \delta_{ij} + \sum_{s=2}^{J} (-1)^{s+1} \sum_{s} \prod_{j \in s} \delta_{ij} < T, \qquad (i=1, \cdots, M). \tag{4.4}$$

In a sequence of vectors which satisfies partial ordering if the first vector with $\delta_{iJ}=0$ doesnot satisfy the tool magazine capacity constraint, so do the other vectors in the range, and we can skip safely to the next uncomparable vector in the solution procedure. However if it satisfies the constraint then try the next vector in the partial ordering, e.g., the first vector with $\delta_{iJ}=1$, apply the same procedure, and we may or may not skip to the next uncomparable vector. If skipping is not permitted, the procedure must continue it's enumeration. This simple rule eliminates considerable amount of states at stage 1 of the dynamic programming procedure, and the result can be applied directly to the following stages without any further computation.

The objective function of (P 3) is to make $\sum_{j=1}^{J} S_j \chi_{ij}$ to be equal to ρ_i^* as much as possible. Also it is convex in the range of sequence which satisfies the partial ordering. Hence it is not necessary to enumerated all the vectors if it diverges much from ρ_i^*. i=1, ⋯, M. Let ξ be a predetermined value and consider a sequence of vectors which is ordered under the partial ordering and satisfy the tool magazine capacity. Then the following can be established. If the value of $\sum_{j=1}^{J} S_j \chi_{ij}$ of the smallest vector under the partial ordering is greater than $\rho_i^* + \xi$ or if the value of the largest vector is less than $\rho_i^* - \xi$, then it is possible to skip to the next uncomparable vector. Otherwise, keep in memory only those vectors with the value of objective function whose objective function value lies within the range of $\rho_i^* \pm \xi$. In other words, at each stage the states are truncated according to their values of the objective function within $\rho_i^* \pm \xi$. Furthermore, at stage k, k=1, ⋯, M, only those values of $\Omega_k(A_{k1}, \cdots, A_{kJ})$ which lie in the range of $\sum_{i=1}^{k} \rho_i^* + \xi$ will be kept in the memory. Also, it should be noticed that, when all the stations have the same number of machines, the results computed at stage 1 can be applied through all the successive stages.

These simple rules considerably reduce the number of states at each stage and accordingly the number of computations. Table 1 summarizes the logical steps following this procedure.

Table 1



the smallest vector in the partial ord-ering
doesnot satisfy tool magazine capacity constraint : skip to the next uncomparable vector
satisfies tool magaz-ine capacity cons-traints : calculate $\sum_j S_j \chi_{ij}$
$\sum_j S_j \chi_{ij} > \rho_i^* + \xi$: skip to the next uncomparable vector
$\sum_j S_j \chi_{ij} < \rho_i^* + \xi$: go to the next vector under par-tial ord ering
does not satisfy tool magazine capacity constraints : skip to the next uncomparable vector
satisfies tool magaz ine capacity cons-traint : calculate $\sum_j S_j \chi_{ij}$ of the largest vector under partial ordering
$\sum_j S_j \chi_{ij} < \rho_i^* - \xi$: skip to the next unco mparable vector
otherwise, do enumeration, keep in memory only those values which lie within $|\sum_j S_j \chi_{ij} - \rho_i^*| < \xi$

5. A Numerical Example

Consider a FMS with 6 machines, grouped into 3 work stations. The number of machines at each station are, respectively, $s_1=3$, $s_2=2$, and $s_3=1$. The number of pallets, N, is 8 which is equivalent to the total part population within the system. The tool magazine capacity at each station(machine), T, is equal to 10.

Then the optimal long-term loading policy can be established, which is not sensitive to fluctuations in the part types and operations mix. To generate result for this loading problem, the algorithm in section 3 is coded in Basic and run on IBM personal computer, and the optimal loading vector, $\underline{\rho}^*$, turns out to be(55.2, 32.6, 12.2) in ratio, which is unique unbalanced loading.

Suppose, in the short-term planning horizon, there are four part types to be manufactured. The number of operations of each type, ω_j, j=1, 2, 3, 4, is respectively, 10, 8, 4, and 3. Also their relative processing times, S_j's, and the numbers of tool slots required, t_j's, are respectively 6, 12, 15, 18, and 3, 4, 5, 7. Then the total workload assigned in the system is 270 and the optimal workloads at stations, ρ_i^*'s, are 149, 88, and 33. Again, the algorithm in section 4 is coded in Basic and run on IBM personal computer, and the optimal short-term workload assignment can be summarized as : $\chi_{12}=7$, $\chi_{13}=4$, $\chi_{21}=6$, $\chi_{24}=3$, $\chi_{31}=4$, $\chi_{32}=1$, and the other χ_{ij}'s are all equal to zero.

6. Conclusion

In this paper, the workload distribution problem in Flexible Manufacturing Systems is formulated and solved from two perspectives, the long term policy and the medium and short-term planning and control. The long-term optimal loading policy is characterized by either balanced loading or unique unbalanced loading. Incorporated with the previous results, we focus on identifying the unique unbalanced loading and develope a steepest ascent method. The optimal loading is then applied to study the optimal medium and short-term planning and control problems. A truncated dynamic programming method is developed which considerably reduces the number of computations to obtain the optimal allocation of the given operations mix of part types to stations.

Reference

1. Bruell, S.C. and G. Balbo, 1980, Computational Algorithms for Closed Queueing Networks(New York : North Holland)
2. Buzacott, J.A., and J.G. Shanthikumar, 1980, Models for Understanding Flexible Manufacturing Systems, AIIE Transactions, 12, 339−350.
3. Buzacott, J.A., and D.D. Yao, 1986, On Queueing Network Models of Flexible Manufacturing Systems. Queueing Systems, Theory and Applications, 1, 5−28.
4. Gill, P. E., Murray, W. and M. H. Wright, 1981, Practical Optimization(New York : Academic Press)
5. Gorden, W.J. and G.F. Newell. 1967, Closed Queueing Networks with Exponential Servers. Operations Research, 15, 252−267.
6. Jackson, J.R., 1963, Jobshop-Like Queueing Systems, Manaement Science, 10, 131−142.
7. Kobayashi, H. and M. Gerlar, 1983, Optimal Routing in Closed Queueing Networks. ACM Transactions on Computer Systems, 1, 294−310.
8. Lawler, E.L. and M.D. Bell, 1967, A Method for Solving Discrete Optimization Problems. Operations Research, 15, 1098−1112.
9. Reiser, M., 1981, Mean-Value Analysis and Convolution Method for Queue-Dependent Servers in Closed Queueing Networks. Performance Evaluation, 1, 7−18.
10. Shanthikumar, J. G., 1983, On the Superiority of Balanced Load in Flexible Manufacturing Systems. unpublished manuscript, University of Syracuse, U.S.A.
11. Shanthikumar, J. G. ad K. E. stecke, 1986, Reducing Work-In-process Inventory in certain Classes of Flexible Manufacturing Systems. European Journal of Operations Research, 26, 266−271.
12. Stecke, K.E., 1983, Formulation and Solution of Nonlinear Integer Production Planning Problems for Flexible Manufacturing Systems, Management Science, 29, 273−288.
13. Stecke, K.E., 1985, The Optimality of Unbalancing Both Workloads and Machine Group Sizes in Closed Queueing Networks of Multi-Server Queues. Operations Research, 33, 882−910.
14. Yao, D.D., 1985, Some Properties of the Throughput Function of the Closed Network of Queues. Operation Research Letters, 3, 313−318.
15. Yao, D.D., 1987, Majorization and Arrangement Orderings in Open Networks of Queues. Annals of Operations Research, 9, 531−543.
16. Yao, D.D., and S.C. Kim, 1987a, Some Order relations in Closed Networks of Queues with Multi-Server Stations. Naval Research Logistics, 34, 53−66.
17. Yao, D.D., and S.C. Kim, 1987b, Reducing Congestion in a Class of Job-Shops. Management Science, 33.