INT. J. PROD. RES., 2001, VOL. 39, NO. 5, 953–979

Machine loading problem of FMS: a fuzzy-based heuristic approach

N. K. VIDYARTHI† and M. K. TIWARI‡*

A fuzzy-based solution methodology has been formulated to address the machine-loading problem in a flexible manufacturing system. The objectives considered are minimization of system unbalance and maximization of throughput, whereas the systems technological constraints are posed by availability of machining time and tool slots. The job ordering/job sequence determination before loading is carried out by evaluating the membership contribution of each job to its characteristics such as batch size, essential operation processing time and optional operation processing time. The operation–machine allocation decisions are made based on the evaluation of membership contribution of operation machine allocation vector. The formulation of membership function is based on logical derivations and enjoys reasonable analytical support. The proposed heuristic is tested on 10 problems adopted from literatures and the results reveal substantial improvement in solution quality over, some of existing heuristic-based approaches.

1. Introduction

The objective of developing FMS is to achieve the efficiency of well-balanced automated high-volume mass production while maintaining the flexibility of low-volume job shop production. The benefits expected of FMS include lower direct manufacturing costs resulting from minimization in set-up time, processing time, lead time, in-process inventory, labour requirements, consistent quality, the ability to respond quickly to design changes, changes in market demand, and adjustment in the event of machine failures, etc. (Singhal *et al.* 1987).

Managing an FMS requires more decisions than that of traditional production line or job shop system. These decisions can be broadly classified into two types: pre- and post-release decisions. FMS planning problem that deals with the pre-arrangement of jobs and tools, before it begins to process, falls under the category of pre-release decisions whereas FMS scheduling problem which considers the sequencing and routing of jobs at the time the system is in operation is a part of post-release decisions. Stecke (1983) described the FMS planning problem as complicated one and divided the same into five subproblems: (1) machine grouping, (2) part type selection, (3) production ratio determination, (4) resource allocation and (5) loading. Here, an attempt has been made to address the machine-loading problem which can be described as, ‘given a set of jobs to be produced, set of tools that are needed for

Revision received May 2000.

† Department of Mechanical Engineering, North Eastern Regional Institute of Science and Technology (NERIST), Itanagar — 791 109, India.

‡ Department of Manufacturing Engineering, National Institute of Foundry and Forge Technology (NIFFT), Hatia, Ranchi, 834 003, India.

* To whom correspondence should be addressed. e-mail: mkt09@hotmail.com

International Journal of Production Research ISSN 0020–7543 print/ISSN 1366–588X online © 2001 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/00207540010010244

processing the jobs on a set of machines, and using a set of resources such as material handling system, pallets and fixtures, and how the jobs be assigned and tools should be allocated so that some measure of productivity is optimized . . .' (Hwang 1986). It can be observed that decisions related to loading problem forms an important link between strategic and operational level decisions in manufacturing and are to be taken tactfully.

In the proposed study, an attempt has been made to address the machine-loading problems of FMS having objectives of minimizing system unbalance and maximizing throughput, while satisfying the constraints viz., available machining time and tool slots. Job ordering/job sequence determination, operation–machine allocations, and reallocation of jobs are the three main constituents of the machine-loading problem. Because of the imprecision and inexactness in dealing with the job ordering/job sequence determination and operation–machine allocation, the concept of fuzzy membership has been adopted here to address the problem. Each operation–machine allocation is judged by its contribution to the objectives and constraints of loading problem to ensure better system performance by framing appropriate fuzzy membership function.

The paper is organised in the following sections: The next section briefly reviews the approaches reported in literature for solving the machine-loading problem of FMS. The background of fuzzy logic approach-based on which the heuristic has been developed is discussed in Section 3. Section 4 deals with the problem environment. The fuzzy logic-based heuristic is developed and reported in Section 5. Illustration of steps of the proposed heuristic has been discussed in Section 6 and complete procedure is explained in the Appendix. The proposed heuristic has been tested on nine more problems and the results are discussed in Section 7. Finally, conclusions are drawn and scope of future research is indicated in Section 8.

2. Literature review

2.1. *Machine loading problem of FMS*

Based on the available literature for solving the loading problem of FMS, the different approaches adopted to solve the machine-loading problem of FMS, can be broadly classified as follows.

(1) Mathematical programming approaches: Stecke (1983), Sawik (1988, 1990, 1996), Sarin and Chen (1987), O'Grady and Menon (1987), Liang and Dutta (1993a, b), Shanker and Tzen (1985), Escudero (1989), Nayak and Acharya (1998).
(2) Multicriteria decision-making approach: Kumar *et al.* (1990), Chen and Askin (1990), Ammons *et al.* (1985).
(3) Simulation-based approach: Jain *et al.* (1989), Kim and Kim (1994), Choi and Malstorm (1988), Gympah and Meredith (1996).
(4) Heuristic-based approach: Mukhopadhyay *et al.* (1991, 1992, 1998), Shanker and Srinivasulu (1989), Moreno and Ding (1993), Tiwari *et al.* (1997), Mukhopadhyay and Tiwari (1995).

Stecke (1983) described six objectives of loading problem in FMS: (1) balancing the machine processing time, (2) minimizing the number of movements, (3) balancing the workload per machine for system of groups of pooled machines of equal sizes, (4) unbalancing the workload per machine for system of groups of pooled machines of unequal sizes, (5) filling the tool magazine as densely as possible, and (6) maximizing the sum of operation priorities. Some of the objectives are contradictory in various situations, while in others several objectives may equally be applicable. The most commonly used loading objective is to balance the workload on all machines, when the machines are not pooled into groups. It has been established that this objective maximizes the expected production (Stecke and Morin 1985). The mathematical formulation of the FMS loading problem was given by Stecke (1983) in which the grouping and loading are formulated as non-linear 0–1 mixed integer programs. Ammons *et al.* (1985) developed a bicriterion objective for the loading problem, i.e. balancing workloads and minimizing visits to the workstations. Shanker and Tzen (1985) also developed a bicriterion objective for the loading problem, i.e. balancing workloads and meeting due dates of part types. Several researchers have considered a wide spectrum of multi-objective loading problems by combining two or more criteria (Kim and Yano 1994, Shanthikumar and Yao 1987, 1988, Dallery and Stecke 1990, Chen and Askin 1990, Tiwari *et al.* 1997). Sawik (1996) proposed a bicriterion integer formulation for machine loading and part routeing with an objective of balancing machine workloads and intermachine flows of parts in the upper level of multilevel decision model for simultaneous machine and vehicle scheduling in a flexible manufacturing system. An integer programming formulation and an interactive solution procedure for a bicriterion loading problem in a flexible assembly system was proposed by Sawik (1997). The objective of the problem was to allocate assembly tasks and products to stations with limited working space, so as to balance the station workloads and minimise station-to-station product transfer line, subject to precedence relations among the tasks for a mix of product types. Sawik (1998) presented a bi-objective integer programming formulation and an approximate lexicographic approach for a bicritrion loading and routing in a flexible assembly system with the objective of balancing station workloads and minimise total inter-station transfer time.

So far the MIP (mixed integer programming) approach for solving the machine-loading problem is concerned, it has been proved to be computationally infeasible even for deterministic formulation. Moreover, the computational time required for solving the loading problem of even a moderate size FMS, is considerably large when the MIP approaches are adopted. This motivated the researchers to develop fast and effective heuristics for solving the loading problem of a large size FMS (Shanker and Srinivasulu 1989, Mukhopadhyay *et al.* 1991, 1992, 1998, Moreno and Ding 1993, Tiwari *et al.* 1997, Mukhopadhyay and Tiwari 1995).

Berrada and Stecke (1986) developed a branch-and-bound algorithm for balancing workloads on machines. Liang and Dutta (1992, 1993a, b) proposed an integrated approach to part selection and machine-loading problems. However, for the simplicity of the problem, most of the researchers have treated job selection, machine loading and tool configuration in a discrete manner, though they are connected by common restrictions such as tool magazine capacity, job tool-machine compatibility and available machining time.

Shanker and Srinivasulu (1989), Mukhopadhyay *et al.* (1992), Tiwari *et al.* (1997) and Moreno and Ding (1993) used fixed predetermined job ordering/job sequencing rule as input to their heuristics proposed for allocation of operations to the machines using minimization of system unbalance and maximization of throughput as objectives subject to constraints posed by availability of machining time and tool slots on machines. In the above heuristic solutions, most of the time, SPT (shortest processing time) has been used as a job ordering rule because it works better on an average as claimed by Stecke and Solberg (1981), Shanker and Tzen (1985), Moreno and Ding (1993), etc. Job ordering has vital impact on the operation–machine allocation decisions in the context of loading problems of FMS.

3. Background of fuzzy logic approach

Job ordering/job sequencing, which is carried out before machine loading in a flexible manufacturing environment, plays a significant role in deciding the allocation of operations on machines. To date, no literature suggests procedure for job ordering determination that directly accounts for the objectives and systems technological constraints of machine-loading problem. To model the realistic operational situations, an attempt has been made in this research to determine the job ordering based on the manufacturing characteristics and other attributes of the jobs to be processed in the system. The batch size of the job, essential operation processing time of job, optional operation processing time of job etc., are some of the attributes which needs to be accounted for while addressing the machine-loading problem. 'Essential operation' of a job refers to the operation that can be performed on a particular machine only, whereas 'optional operation' refers to the operation that can be performed on alternative machines. In real manufacturing situation, the information related to aforementioned attributes is imprecise and conflicting in nature.

Moreover, when the operation machine allocation is under taken, the objectives can be imprecise and conflicting in nature. For example, consider an FMS having four machines M1–4, each having 480 min of processing time and five tool slots in a given planning horizon. The description of the job *j* to be allocated on to the machines are as: operation 1 on M1 (200 min, two slots), operation 2 on M1 (250 min, two slots) or M2 (300 min, three slots) or M3 (325 min, five slots) or M4 (360 min, six slots) and operation 3 on M2 (110 min, one slot) and operation 4 on M1 (150 min, one slot). Operations 1, 3 and 4 are essential, while operation 2 is optional as it can be allocated to M1, M2, M3 or M4. There are four sets of operation machine allocations for this job. Each set consists of the description of the operations that is to be assigned to a particular machine along with processing time and tool slot requirements. Set $1 = \{M1(200,2), M2(300,3), M2(110,1), M1(150,1)\}$ denotes that the operation 1 of job *j* is to be allocated to M1 needing 200 min of processing time and two tool slots, operation 2 of the job *j* is to be allocated to M2 needing 300 min and three tool slots, operation 3 on M2 with 110 min of processing time and one tool slot. Similarly, other sets of operation machine allocations for this job *j* are: set $2 = \{M1(200,2), M3(325,5), M2(110,1), M!(150,1)\}$, set $3 = \{M1(200,2), M4(360,6), M2(110,1), M1(150,1)\}$, and set $4 = \{M1(200,2), M1(250,2), M2(110,1), M1(150,1)\}$. If set 1 is selected, and the operations are allocated, the time remaining on machine 2 is not sufficient to meet the processing requirements (essential operations) of the other unassigned jobs. If set 2 is selected, it leaves behind insufficient tool slots on machine 3, and hence all the jobs needing machine 3 are to be left unassigned. If set 3 is selected, it violates the tool slot constraints on machine 4. If set 4 is selected, then M1 is to be overutilized. On one hand, set 1 is better than set 2, as the chance of tool slot constants violations are minimum while allocating unassigned jobs, whereas set 2 is better than set 1 if unassigned jobs do not need machine 3 (or have alternative machine options). Also set 4 is better than set 2 as it tends to minimize system unbalance. In this situation, it is very hard to make a choice of sets of operation machine allocation since the objectives are conflicting, information is imprecise, and multiple parameters are to be handled simultaneously.

Fuzzy logic is a powerful tool to deal with imprecise information, and to handle vague concepts. By applying fuzzy logic, we can quantify the contribution of set of operation–machine allocation to the various parameters in terms of fuzzy membership. The foundation of fuzzy logic is fuzzy set theory. Zadeh (1965) first introduced fuzzy set theory. Unlike classical logic, fuzzy logic is aimed at providing a body of concepts and techniques for dealing with modes of reasoning which are approximate rather than exact. Fuzzy set theory uses a multivalued membership function to represent membership of an object in a class rather than using the classical binary. Zero or one values are used to denote membership. If A be a classical set of objects whose elements are denoted by x, then the membership in a classical subset of X of A is often viewed as characteristic function of μ_X from A to a valuation set $\{0, 1\}$, such that

$$\mu_X(x) = \begin{cases} 1 & \text{if and only if } x \in X \\ 0 & \text{otherwise,} \end{cases}$$

where X is called a fuzzy set if the valuation set is allowed to be in the real interval of [0, 1] and $\mu_X(x)$ is the grade of membership of x in X. The higher $\mu_X(x)$, the more x belongs to X. For more detail and better understanding of fuzzy set theory, see Zimmerman (1991, 1992), Dubois and Prade (1980) and Ross (1997).

As reported by Zadeh (1991), fuzzy logic has diversified applications ranging from medical diagnosis and investment management to consumer electronics and industrial process control. Since, Bellman and Zadeh (1970) introduced the concept of fuzzy decision-making, various applications of fuzzy theory to decision-making problems have been presented. They pointed that in a fuzzy environment, goals and constraints normally have the same nature and can be represented by fuzzy sets on A, where A is the set of alternatives that contains the solution of a given multicriteria optimization problem. Zhang and Huang (1994) have used a fuzzy-based approach to solve process plan selection problem by evaluating membership function to various objectives that were imprecise and conflicting in nature. Yu *et al.* (1999) presented a fuzzy interference-based scheduling decision for FMS with multiple objectives. In their example, the FMS processes five products and each product needs two manufacturing operations on three machines. The operation sequence for each product is known in advance but there is a choice for operations for their processing on machines. The fuzzy inference-based methodology is employed in their research to decide the assignment of jobs to suitable machines and ordering of all jobs assigned to a given machine.

Mukhopadhyay (1999) observed that a typical machine-loading problem in FMS is not always predictable owing to the fuzzyness in the system behaviour and adopted the concept of fuzzy entropy measurement to resolve the loading problem. Singh and Mohanty (1991) adopted a fuzzy approach to a multi-objective routing problem and applied the model to resolve a process plan selection issues encountered in a manufacturing environment.

Kazerooni *et al.* (1997) adopted a fuzzy approach to real time operation selection in an FMS and used simulation methods to demonstrate the effectiveness of their approach. Noto La Diega *et al.* (1995) developed a fuzzy programming model to determine the cutting speeds and the volume ratios of each part type processed in a

FMS environment. In this research, attempt has been made to extend these concepts for solving machine-loading problem of FMS.

4. Problem environment

This study considers a random type FMS, capable of producing several kinds of products for which orders arrive in a random manner: each order stands for one product type; the product may require several operations and may have alternative routings, i.e. several types of machine may be capable of processing the same operation, and the system may comprise of several machines of same type. A random FMS has been considered rather than a dedicated type FMS, the reason being, a dedicated type system is designed to produce a rather small family of similar parts with a known and limited variety of processing needs whereas a random FMS is designed for a large family of parts having a wide range of variations in characteristics. Unlike dedicated FMS, in a random FMS, loading decisions are dynamic and thus are made for a specified planning horizon. Jobs are to be sequenced and processed over a given planning horizon. The loading problem addressed in this paper is that, although machine capacity might be sufficient, it may not be possible to process all the job orders required in a particular planning period due to limited number of tool slots and available machining time. Thus, a subset of job orders has to be selected and the required tools allocated to the machine before the orders are to be processed. The system has a number of versatile machines because of which simplification of problem can not be ensured. A job has one or more operations and each operation can be performed by one or more machines.

The processing time and tool slots required for each operation of the job and its batch size are known before hand. Operation that has to be performed on a particular machine with certain number of tool slots is termed as essential operation, whereas optional operations are those which can be processed on alternative machines (with same or different processing time and tool slots). The FMS under consideration derives its flexibility in selection of machine for optional operation of the job. Shanker and Tzen (1985) have proposed a test problem, by capturing the above features of FMS, which is given in the table 1. Mukhopadhyay *et al.* (1992) generated nine more such test problems to mimic the real life loading scenarios. Owing to flexibility in operation routings, the possible number of operation allocation on machines of a random FMS, along with variations in job sequences turns out to be very large. For example, consider the problem given in table 1, where there are eight jobs with 8! job sequencings. For one order, there exists 2592 operation–machine allocation combination ($= 1 \times 2 \times 2 \times 1 \times 2 \times 6 \times 9 \times 6$). Therefore, for 8! job orderings, total number of allocations comes out to be 104 509 440. Determination of near-optimal solution in such a large search space is a computationally intractable task and needs some heuristic approach to arrive at near-optimal solution. Though, heuristic approaches do not guarantee optimal solution but it reduces the exhaustive search procedure.

In this research, artificial intelligence-based heuristic solution methodology has been proposed to solve the loading problem. Like most of the heuristics developed so far (Mukhopadhyay *et al.* 1992, Shanker and Srinivasulu 1989, Tiwari *et al.* 1997), the proposed heuristic adopts 'minimization of system unbalance' and 'maximization of throughput' as two common yardsticks to check the optimality of allocation in presence of availability of machine time and tool slots, as systems technological constraints. System unbalance has been defined as the sum of over- or underutilized

Job number	Operation number	Batch size	Unit processing time	Machine number	Slot needed
1	1	8	18	3	1
2	1	9	25	1	1
	2		24	4	1
	3		22	2	1
3	1	13	26	4	2
	2		11	3	3
4	1	6	14	3	1
	2		19	4	1
5	1	9	22	2	2
	2		25	2	1
6	1	10	16	4	1
	2		7	4	1
				2	
				3	
	3		21	2	1
7	1	12	19	3	1
				2	
				4	
	2		13	2	1
				3	
				1	
	3		23	4	3
8	1	13	25	1	1
				2	
				3	
	2		7	2	1
				1	
	3		24	1	3

Table 1. Job description. [Taken from Shanker and Tzen (1985).]

time on the machines available in the system. Throughput refers to the unit of jobs processed. Minimization of system unbalance is equivalent to the maximization of total system workload. Most of the researchers have addressed the machine-loading problem using two objectives, i.e. workload balance and throughput. Since, the FMS involves high capital investment, the investor aims to achieve high machine utilisation which is related to minimisation of idle time. This is achieved by balancing the workload. Moreover, FMS are usually employed for medium production volume and medium part variety. The throughput of the system is an important and primary objective and cannot be overlooked. Moreover, while attempting to maximise throughput by balancing system workloads often has the side benefit of limiting tardiness (Kim and Yano, 1997).

However, to minimize the complexities in analysing the problem for a practical FMS, following assumptions have been made.

(1) Unique job routing.
(2) Non-splitting of jobs.
(3) No sharing and duplication of tools.
(4) Availability of resources such as pallets, fixtures and AGV, etc. is in sufficient numbers.

5. Proposed fuzzy logic-based solution methodology

5.1. *Notation*

The following notation is used:

j	job number, $j = 1,2,3,\ldots,J$,
m	machine number, $m = 1,2,3,\ldots,M$,
k	operation number, $k = 1,2,3,\ldots,K$,
n	operation–machine allocation vector, $n = 1,2,3,\ldots,N$,
q_j	batch size for the job j,
ute_{kjm}	unit processing time out the essential operation k of job j on machine m,
uto_{kjm}	unit processing time for carrying out the optional operation k of job j on optional machine m,
te_{kjm}	total processing time for carrying out the operation k of job j on machine $m = q_j \times \mathrm{ute}_{kjm}$,
to_{kjm}	total processing time for carrying out the optional operation k of job j on machine $m = q_j \times \mathrm{uto}_{kjm}$,
se_{kjm}	tool slots required for carrying out the essential operation k of job j on machine m,
so_{kjm}	tool slots required for carrying out the optional operation k of job j on machine m.
ET_{mj}	essential time requirement on machine m after allocation of essential operations of job j; $$\mathrm{ET}_{mj} = \sum_{j=1}^{J}\sum_{k=1}^{K} \mathrm{te}_{kjm},$$
ES_{mj}	essential tool slot requirement on machine m after allocation of essential operation of job j,
ET'_{mj}	essential time requirement on machine m before allocation of essential operation of job j: $$\mathrm{ES}_{smj} = \sum_{j=1}^{J}\sum_{k=1}^{K} \mathrm{se}_{kjm},$$
ES'_{mj}	essential tool slot requirement on machine m before allocation of essential operation of job j,
ET_m	essential time requirement on machine m,
ES_m	essential tool slot requirement on machine m,
T_m	available machine time on machining m at $t = 0$,
S_m	available machine tool slots on machine m at $t = 0$,
RT_m	remaining (updated) time on machine m,
RS_m	remaining (updated) tool slots on machine m,

AT_m	available time on machine m,
AS_m	available tool slot on machine m,
OMAV_n^j	nth operation machine allocation vector of job j: it is a vector representing a set of operations (essential and optional) required for job j,
$\mathrm{RT}_m[\mathrm{OMAV}_n^j]$	remaining time on machine m after the allocation of operations of job j as per nth operation machine allocation vector,
$\mathrm{RS}_m[\mathrm{OMAV}_n^j]$	Remaining tool slot on machine m after the allocation of operations of job j as per nth operation machine allocation vector,
S	set of jobs arranged in order for loading on machines,
U	set of unassigned jobs,
A	set of assigned jobs,
μ_e^j	membership function for the essential processing time of job j,
μ_o^j	membership function for the optional processing time of job j,
μ_q^j	membership function for the batch size of job j,
w_e, w_o and w_q	weights assigned to the individual membership functions $\mu_e^j, \mu_o^j, \mu_q^j$ respectively ($w_e = w_o = w_q = 1$ for the case under consideration),
SU_A	system unbalance for the assigned jobs: $= \sum_{m=1}^{M} \mathrm{RT}_m$,
SU	final system unbalance,
Th_A	throughput corresponding to the set of assigned jobs,
μ_s^j	overall membership function of job j for arranging them in the order: $= \sum_{j=1}^{J} q_j, \quad \forall j \in \mathrm{A}.$

5.2. *Definitions*

- μ_e^j: The membership function for the essential processing time of job j is defined as the ratio of difference between the maximum of total essential processing time of the jobs and the essential processing time of job j to the difference between the maximum and minimum of total essential processing time of jobs. This can be expressed as:

$$\mu_e^j = \frac{\left(\sum_{k=1}^{K} \mathrm{te}_{kjm}\right) \max - \left(\sum_{k=1}^{K} \mathrm{te}_{kjm}\right)}{\left(\sum_{k=1}^{K} \mathrm{te}_{kjm}\right) \max - \left(\sum_{k=1}^{K} \mathrm{te}_{kjm}\right) \min} \tag{1}$$

$$\left(\sum_{k=1}^{K} \mathrm{te}_{kjm}\right) \max = \text{maximum of} \sum_{k=1}^{K} \mathrm{te}_{kjm} \quad \forall j = 1,2,3,\ldots,J$$

$$\left(\sum_{k=1}^{K} \mathrm{te}_{kjm}\right) \min = \text{maximum of} \sum_{k=1}^{K} \mathrm{te}_{kjm} \quad \forall j = 1,2,3,\ldots,J$$

$$0 \le \mu_e^j \le 1.$$

- μ_o^j: The membership function for the optional processing time of job j is defined as the ratio of difference between the minimum of total processing time of optional operations of jobs and the processing time of optional operations of job j to the difference between the maximum and minimum of total processing time of optional operations of the jobs. This can be represented as:

$$\mu_o^j = \frac{\left(\sum_{k=1}^{K} \mathrm{to}_{kjm}\right) - \left(\sum_{k=1}^{K} \mathrm{to}_{kjm}\right) \min}{\left(\sum_{k=1}^{K} \mathrm{to}_{kjm}\right) \max - \left(\sum_{k=1}^{K} \mathrm{to}_{kjm}\right) \min} \tag{2}$$

$$\left(\sum_{k=1}^{K} \mathrm{to}_{kjm}\right) \max = \text{maximum of} \sum_{k=1}^{K} \mathrm{to}_{kjm} \quad \forall j = 1,2,3,\ldots,J$$

$$\left(\sum_{k=1}^{K} \mathrm{to}_{kjm}\right) \min = \text{maximum of} \sum_{k=1}^{K} \mathrm{to}_{kjm} \quad \forall j = 1,2,3,\ldots,J$$

$$\sum_{k=1}^{K} \mathrm{to}_{kjm} = \text{sum of processing time of all the optional operations of job } j.$$

$$0 \le \mu_o^j \le 1.$$

- μ_q^j: The membership function for the batch size of job j is defined as the ratio of difference between the minimum of batch size of jobs and the batch size of job j to the difference between the maximum and minimum of batch size of jobs. This can be written as:

$$\mu_q^j = \frac{(q_j) - (q_j) \min}{(q_j) \max - (q_j) \min} \tag{3}$$

$$(q_j) \max = \text{maximum of } q_j, \quad \forall j = 1,2,3,\ldots,J$$

$$(q_j) \min = \text{minimum of } q_j, \quad \forall j = 1,2,3,\ldots,J$$

$$0 \le \mu_q^j < 1.$$

- μ_s^j: The overall membership function of job j is the weighted sum of the individual membership function of job j to the essential operation processing time, optional operation processing time and batch size respectively. This can be expressed as:

$$\mu_s^j = \frac{W_e \mu_e^j + W_o \mu_o^j + W_q \mu_q^j}{W_e + W_o + W_q} \tag{4}$$

$$0 \leq \mu_s^j \leq 1.$$

The jobs are arranged in descending order of value of μ_s^j.

- OMAV_n^j denotes the nth set of operation-machine allocation vector of jth job and is defined as a vector representing a set of operations (essentail and/or optional) required for job j.

  For example, if job 1 has three operation; first operation is essentail operation with a requirement of 230 min and three slots on M1, second operation is optional operation with a requirement of 300 min and two tool slots on M3 or M2, third operation is optional operation with a requirement of 250 min and two slots on M1 or M2, then

$$\mathrm{OMAV}_1^1 = \{\mathrm{M1}(230,3), \mathrm{M3}(300,2), \mathrm{M1}(250,2)\}$$

$$\mathrm{OMAV}_2^1 = \{\mathrm{M1}(230,3), \mathrm{M2}(300,2), \mathrm{M2}(250,2)\}$$

$$\mathrm{OMAV}_3^1 = \{\mathrm{M1}(230,3), \mathrm{M3}(300,2), \mathrm{M2}(250,2)\}$$

$$\mathrm{OMAV}_4^1 = \{\mathrm{M1}(230,3), \mathrm{M2}(300,2), \mathrm{M1}(250,2)\}.$$

- $\mathrm{RT}_m[\mathrm{OMAV}_n^j]$: If job j is the first element of set S, then the remaining time on machine m after the allocation of operations of job j, as per nth operation machine allocation vector is given by:

$$\mathrm{RT}_m[\mathrm{OMAV}_n^j] = \mathrm{T}_m - \sum_{k=1}^{K} \mathrm{te}_{kjm} - \sum_{k=1}^{K} \mathrm{to}_{kjm}$$

  and AT_m before allocation of operations of $(j+1)$th job, where $(j+1)$th job is the second element of set S, is equal to $\mathrm{RT}_m[\mathrm{OMAV}_n^j]$.

  If job j is not the first element of sets, then

$$\mathrm{RT}_m[\mathrm{OMAV}_n^j] = \mathrm{AT}_m - \sum_{k=1}^{K} \mathrm{te}_{kjm} - \sum_{k=1}^{K} \mathrm{to}_{kjm}.$$

- RS_m $[\mathrm{OMAV}_n^j]$: If job j is the first element of set S, then the remaining tool slots on machine m after the allocation of operations of job j as per nth operation machine allocation vector is given by:

$$\mathrm{RS}_m[\mathrm{OMAV}_n^j] = \mathrm{S}_m - \sum_{k=1}^{K} \mathrm{se}_{kjm} - \sum_{k=1}^{K} \mathrm{so}_{kjm}$$

  and AS_m before allocation of operations of $(j+1)$th job, where $(j+1)$th job is the second element of set s is equal to $\mathrm{RS}_m[\mathrm{OMAV}_n^j]$.

  If job j is not the first job in the sequence, then,

$$\mathrm{RS}_m[\mathrm{OMAV}_n^j] = \mathrm{AS}_m - \sum_{k=1}^{K} \mathrm{se}_{kjm} - \sum_{k=1}^{K} \mathrm{so}_{kjm}.$$

- $\mu[\mathrm{OMAV}_n^j]$: Membership function of nth operation-machine allocation vector of job j

$$
= \frac{1}{M}\sum_{m=1}^{M}\left\{\frac{\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_n^j]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)_m-\min\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_n^j]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)}{\max\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_n^j]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)-\min\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_n^j]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)}\right\}
$$

$$
\times \frac{1}{M}\sum_{m=1}^{M}\left\{\frac{\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_n^j]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)_m-\min\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_n^j]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)}{\max\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_n^j]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)-\min\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_n^j]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)}\right\} \tag{5}
$$

is denoted by $\mu[\mathrm{OMAV}_n^j]$ and is expressed as:

$$= 0 \text{ if } \mathrm{RS}_m[\mathrm{OMAV}_n^j] < 0 \quad \text{for} \quad m = 1, 2, \ldots, M.$$

$$\frac{\mathrm{RS}_m[\mathrm{OMAV}_n^j]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m} = 1, \quad \text{if } \mathrm{RS}_m[\mathrm{OMAV}_n^j] = \mathrm{AS}_m$$

and

$$\frac{\mathrm{RS}_m[\mathrm{OMAV}_n^j]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m} = 0, \quad \text{if } \mathrm{RS}_m[\mathrm{OMAV}_n^j] \neq \mathrm{AS}_m \text{ but}, \mathrm{AS}_m = \mathrm{ES}_m$$

$$\frac{\mathrm{RT}_m[\mathrm{OMAV}_n^j]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m} = 1, \quad \text{if } \mathrm{RT}_m[\mathrm{OMAV}_n^j] = \mathrm{AT}_m$$

and

$$\frac{\mathrm{RT}_m[\mathrm{OMAV}_n^j]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m} = 0, \text{ if } \quad \mathrm{RT}_m[\mathrm{OMAV}_n^j] \neq \mathrm{AT}_m \text{ but}, \ \mathrm{AT}_m = \mathrm{ET}_m.$$

These have been to avoid division by zero in various situations arising during evaluation of membership.

5.3. *Proposed solution methodology*

In this research, an attempt has been made to use an intelligent heuristic to capture the intricate details of loading problems of an FMS in a wider environment, where machines are capable of performing several types of operations, an operation can be performed on several alternative machines using several tool types and a job may have alternative processing routes. In the proposed methodology, the sequencing problem has been solved before machine-loading problem. The jobs in a given batch size are sequenced by evaluating the overall contribution of the attributes such as batch size, essential operations processing time, optional operations processing time. As per the sequence, the operations of the jobs are allocated onto the machines by satisfying machining time and tool slots constraints of the system. If any of the operations of the job is left unassigned, because of system technological constraints, then the job is eliminated from the sequence. After all the operations-machine allocations are done, a set of selected jobs and their loading sequence is obtained. Based on this, the scheduling decisions can be made. Hence, job sequencing before machine loading simplifies the burden of job selection and this will help in dissolving the complications encountered in generating efficient schedules. The proposed heuristic focuses on three segments on machine-loading problem: (1) job sequencing determination, (2) operation–machine allocation decisions and (3) reallocation of jobs to meet system constraints.

5.3.1. *Determination of job sequencing*

The job sequencing is determined by evaluating the overall contribution of the fuzzy membership function of the job to the attributes. When a job is considered for assignment on machine, preference is given for a job that consumes the least amount of available resources, such as the number of tool slots and processing time on the machines. If the system workload balance is the primary objective, then the machine with most remaining capacity is considered whereas in case of throughput as primary objective, preference is given to the least possible combination of resources. Because of this conflicting nature, fuzzy membership function has been formulated. The overall membership contribution of a job is the weighted sum of individual membership of job to its attributes such as essential operation processing time, optional operation processing time and batch size respectively. Fuzzy membership functions have been built up to evaluate the individual contribution of a job to an attribute. When essential operation processing time is considered, the job with minimum essential operation processing time has membership value of 1 and the job with maximum essential operation processing time has membership value 0. When optional operation processing time is considered, the job with maximum optional operation processing time has membership value of 1 and the job with minimum optional operation processing time has membership value of 0. When batch size is considered, the job with maximum batch size had membership 1, and that with minimum batch size has membership value 0. These functions are given by equations (1–3) respectively. The overall membership function of the job is given by equation (4). The sequence is determined by arranging the jobs in the descending sequence of overall membership value. These functions have been devised to take into account the following factors.

(1) Jobs with lower essential operation processing time will lie earlier in the sequence, so that allocation of these jobs will leave behind sufficient time on machines to facilitate optional operation machine allocation.
(2) Jobs with higher optional operation time lie earlier in the sequence, so that while allocating these jobs, system constraints are not violated.
(3) Jobs with higher batch size takes earlier position in the sequence, so that these jobs are assigned on machines with less chance of rejection due to violation of system constraints.

5.3.2. *Operation–machine allocation decisions*

For a job, the operation–machine allocation decisions are made by evaluating the membership of all the operation–machine allocation vectors of that job. The membership function of the operation–machine allocation vector has been constructed in such a way that a vector with higher membership value attempts to select a particular combination of allocation of all the operations of the job which leaves behind higher remaining time and higher number of tool slots on the machines so as to

(1) facilitate the allocation of operation of unallocated jobs;
(2) meet the essential time requirement as well as essential tool slot requirement on the machines for processing the other essential operation of jobs that are yet to be allocated. If any operation of the operation machine allocation vector is allocated by violating the tool slot constraints, then its membership value becomes zero. The membership function is given by equation (5);
(3) reallocation of jobs: In sequence to ensure minimum positive system unbalance and maximum possible throughput, the heuristic has provision of reallocation of jobs from the set of unassigned jobs by removing jobs from the set of assigned jobs.

5.4. *Proposed heuristic*

The above mentioned concepts for solving the machine-loading problem have been summarized in the form of various steps of an heuristic, given as follows.

Step 1. Input the number of jobs J, number of machines M, total available processing time on each machine, total available tool slot on each machine, batch size of each job. Also input the unit processing time, machine number, slots needed for every operation (essential and optional) k of job j.
Step 2. Determine the essential time requirement, essential tool slots requirement, available machine time and available tool slot on each machine.
Step 3. For every job j, find $\sum_{k=1}^{K} \mathrm{te}_{kjm}, \sum_{k=1}^{K} \mathrm{to}_{kjm}$ and q.
Step 4. Evaluate μ_e^j, μ_o^j and μ_q^j for $j = 1, 2 \ldots, J$.
Step 5. Evaluate μ_s^j, and determine the job sequencing by arranging job j in the descending sequence of value of μ_s. Place them in the set S. (Use equation 4.)
Step 6. Assign jth job from the sequence in set S:

Step 6.1. Initialize $j = 1$.
Step 6.2. Find all the OMAV_n^j of job J.
Step 6.3. Find RT_m, RS_m, ET_m, ES_m, AT_m, AS_m for every m after allocation of every operation corresponding to OMAV_n^j.
Step 6.4. Evaluate $\mu[\mathrm{OMAV}_n^j]$ for $n = 1 - N$ (using equation 5).
Step 6.5. Choose OMAV_n^j such that $\mu[\mathrm{OMAV}_n^j]$ is maximum and allocate operations as per this vector.

If max $\{\mu[\mathrm{OMAV}_n^j]\}$ is same for more than one operation–machine allocation vector, then select OMAV_n^j that has RT_m, RS_m comparatively higher than ET_m and ES_m respectively.

If $\mu[\mathrm{OMAV}_n^j] = 0$, because of any $\mathrm{RS}_m[\mathrm{OMAV}_n^j] < 0$, then leave that operation machine allocation vector. This is required to prevent the selection of OMAV that violates the tool slot constraints (as there is no flexibility with overloading of tool slots on machines). $\mu[\mathrm{OMAV}_n^j]$ is zero, for $n = 1 - N$, then, reject the job j.

Step (6.6) Update ET'_m, ES'_m, AT_m and AS_m.
Step (6.7) If $j \in J$, then increase j by one and go to Step 6.2 else go to Step 7.

Step 7. Determine the elements of set U and set A.

$$\text{Find } \mathrm{SU_A} = \sum_{m=1}^{M} \mathrm{RT}_m, \quad \text{and} \quad \mathrm{Th_A} = \sum_{j=1}^{J} q_j \quad \text{and} \quad J \in \mathrm{A}.$$

Step 8. If $\mathrm{SU_A}$ is positive, then $\mathrm{SU} = \mathrm{SU_A}$ and $\mathrm{Th} = \mathrm{Th_A}$, otherwise go for reallocation.

Step 9. Reallocation steps:

The reallocation steps have been considered to ensure minimum positive system unbalance and to exploit the possibility of further improvement in the solution quality.

Step 9.1. Find $\mathrm{SU}_{\mathrm{A}j}$, where $\mathrm{SU}_{\mathrm{A}j}$ is the system unbalance obtained after removing the job j from Set A. If the system unbalance is still negative then start remaining pair of jobs from set A. Form the pairs of the jobs in sequence to make system unbalance positive based on the total processing time of the jobs. Select $\mathrm{SU}_{\mathrm{A}j}$ having minimum positive value and determine corresponding throughput.

Step 9.2. Update RT_m, RS_m, ET'_m, ES'_m, A and U.

Step 9.3. Find $\mathrm{SU}_{\mathrm{u}j}$, where $\mathrm{SU}_{\mathrm{u}j}$ is the system unbalance obtained after the allocation of jth job from set U, satisfying systems technological constraints. Find $\mathrm{SU}_{\mathrm{u}j}$ for different combination of allocation of jobs from set U.

Step 9.4. Select the $\mathrm{SU}_{\mathrm{u}j}$ with a minimum positive value and find the corresponding throughput. If the system unbalance is still negative, then update A and U, RT_m, RS_m, ET_m and ES_m and go to Step 9.1, otherwise find values of SU and corresponding Th.

6. Numerical illustration

The proposed heuristic is applied to solve the example problem (table 1), by Shanker and Tzen (1985). A planning period of 480 min has been considered. Nine more test problems (adopted from Mukhopadhyay *et al.* 1992) have been solved to test the computational performance of the proposed heuristic, results of which are discussed in the next section. The job arrivals have been generated assuming an independent and identically distributed exponential distribution with a mean of 70 min for the interarrival time for the set of test problems. In sequence to illustrate the various steps of the heuristic, test problem number 1 (given in table 1) has been considered in detail and is given in the appendix.

7. Results and discussion

The proposed fuzzy logic-based heuristic has been tested on 10 set of test problems (adopted from Mukhopadhyay *et al.* 1992). Table 2 lists the operation machine allocation vector for the jobs assigned for these problems. The results are summarized in table 3, along with job sequencing, set of job assigned and unassigned, system unbalance and throughput. In table 4, a comparison has been made among the results obtained using the proposed heuristic and that offered by heuristic approaches of Shanker and Srinivasulu (1989), Mukhopadhyay *et al.* (1992) and Tiwari *et al.* (1997). The heuristic developed provides better solution to the loading problem and the extent of improvement can be witnessed from table 4.

Some of the important features of the heuristic approach are discussed as below.

Problem no.	Job no.	$OMAV_n^j$
1	3	M1 (338, 2) + M3 (143, 3)
	8	M2 (325, 1) + M2 (91, 1) + M1 (312, 3)
	6	M4 (160, 1) + M2 (70, 1) + M2 (210, 1)
	1	M3 (144, 1)
2	5	M4 (480, 2) + M1 (160, 1) + M1 (256, 1)
	1	M2 (180, 2)
	6	M2 (231, 3)
	4	M3 (225, 1)
	3	M1 (264, 3)
3	3	M4 (440, 1) + M3 (462, 2)
	1	M3 (250, 3)
	2	M2 (310, 3)
	5	M1 (160, 1) + M2 (170, 1)
4	2	M3 (81, 1) + M3 (54, 2)
	4	M2/M3 (168, 2) + M3 (144, 1) + M2 (144, 1)
	5	M1 (240, 2)
	3	M4 (96, 1)
	1	M2 (60, 1) + M1 (72, 2) + M4 (42, 1)
5	4	M4 (140, 1) + M1 (364, 2) + M3 (224, 1)
	1	M2 (144, 2) + M2 (108, 1)
	6	M1 (216, 2)
	5	M3 (240, 3)
	2	M2 (220, 2)
6	5	M4 (480, 2) + M4 (256, 1)
	1	M2 (180, 2)
	4	M3 (225, 1)
	3	M1 (264, 3)
	6	M2 (231, 3)
7	2	M4 (150, 1) + M1 (375, 2) + M3 (240, 1)
	6	M1 (120, 2) + M2 (108, 1)
	4	M3 (210, 1) + M4 (300, 2)
	1	M1 (240, 2)
8	5	M1 (364, 2) + M3 (154, 3)
	6	M3 (325, 1) + M2 (221, 1) + M1 (312, 3)
	7	M2 (70, 1)
	2	M3 (160, 1)
	4	M3 (168, 1) + M4 (133, 1)
9	2	M3 (180, 1)
	5	M4 (80, 1) + M1 (100, 1)
	7	M2 (72, 1) + M4 (81, 2)
	4	M1 (99, 1) + M2 (121, 2)
	1	M3 (150, 2) + M1 (180, 2)
	3	M2 (240, 2)
	6	M4 (176, 1) + M1 (132, 1)
10	5	M1 (325, 1) + M2 (91, 1) + M2 (360, 1)
	1	M4 (91, 1) + M4 (104, 1) + M2 (117, 2)
	2	M3 (81, 1) + M4 (72, 1) + M1 (72, 1)
	6	M3 (168, 2) + M1 (228, 1)
	3	M1 (99, 1) + M2 (90, 1).

Table 2. Operation–machine allocation vectors selected for the jobs assigned.

Problem number	Job sequence	Job assigned (A)	Job unassigned (U)	System unbalance (SU)	Throughput (Th)
1	{3,8,7,6,5,1,2,4}	{3,8,6,1}	{5,7,2,4}	127	44
2	{5,1,6,4,3,2}	{5,1,6,4,3}	{2}	124	63
3	{3,1,2,5,4}	{3,1,2,5}	{4}	128	73
4	{2,4,5,3,1}	{2,4,5,3,1}	{*f*}	819	51
5	{4,1,3,6,5,2}	{4,1,6,5,2}	{3}	264	61
6	{5,1,4,3,6,2}	{5,1,4,3,6}	{2}	284	63
7	{2,6,4,1,5,3}	{2,6,4,1}	{3,5}	177	54
8	{5,6,7,1,2,3,4}	{5,6,7,4}	{3,2,1}	13	44
9	{2,5,7,4,1,3,6}	{2,5,7,4,1,3,6}	{*f*}	309	88
10	{5,1,2,6,4,3}	{5,1,2,6,3}	{4}	122	56

Table 3. Summary of results of the proposed heuristic.

Shift no.	Total no. of jobs	Shanker and Srinivasulu (1989)		Mukhopadhyay *et al.* (1992)		Tiwari *et al.* (1997)		Proposed heuristic	
		SU	Th	SU	Th	SU	Th	SU	Th
1	8	253	39	122	42	76	42	127	44
2	6	388	51	202	63	234	63	124	63
3	5	288	63	286	79	152	69	128	73
4	5	819	51	819	51	819	51	819	51
5	6	467	62	364	76	264	61	264	61
6	6	548	51	365	62	314	63	284	63
7	6	189	54	147	66	996	48	177	54
8	7	459	36	459	36	158	43	13	44
9	7	462	79	315	88	309	88	309	88
10	6	518	44	320	56	166	55	122	56
Average		439.1	53	339.9	61.9	348.8	58.3	236.7	59.7

Table 4. Comparison of the proposed fuzzy logic-based heuristic with the heuristic developed by Mukhopadhyay *et al.* (1992), Tiwari *et al.* (1997) and Shanker and Srinivasulu (1989).

(1) While determining the job sequencing, weights are to be assigned to the individual membership function in sequence to stimulate the real-shop floor situations in varying production scenarios.
(2) Operation–machine allocation decisions are made by evaluating the membership value of all the operation machine allocation vector of a job. In case of machine disruption or failure, the next set of operation–machine allocation vector can be selected to meet the exigencies at shop floor level.
(3) The membership function (given by equation 5) of operation machine allocation vector takes care of both the factors, i.e. feasibility and optimality of allocation. It takes into account various parameters such as: available machining time, available tool slots on machines and future requirements of time and tool slots for carrying out the essential operations of the unassigned jobs at the time the operation machine allocation decisions are to be made. Thus, it acts as a crucial link between tactical planning and operational decisions of the system.

(4) Reallocation procedure in the proposed heuristic is an attempt to minimize the overutilization of system at the cost of minimum possible reduction in system throughput.

8. Conclusion and future scope

The primary contribution of this research is the development of a fuzzy logic-based heuristic approach for the problem of allocating the set of operations of jobs to the machines so as to minimize system unbalance and maximize throughput while satisfying available machining time and tool slot constraints. Unlike other researches, in this study, the procedure of determining the job sequencing followed is unique and aims to meet the objectives and constraints of the system. It has been observed that by adopting such job sequencings, the possible inconsistency posed by system constraints during operation–machine allocation is minimized.

Further efforts may be put forth to extend the methodology for incorporating following details.

(1) Constraints pertaining to availability of jigs, fixtures, pallets can be included in the membership function for operation machine allocation with same objective function.
(2) Minimization of number of late parts can be ensured in the present heuristic by including due date related measures in the form of membership function while determining the job sequencing.
(3) Splitting of job may be allowed to simulate the real-world situations.

Acknowledgements

The authors express their sincere thanks and gratitude to the Editor and anonymous referees for constructive comments and valuable suggestions on an earlier version of the paper. The authors benefited immensely from discussions with Professor S. P. Dutta of University of Windsor, Professor S. K. Mukhopadhyay of NITIE, Mumbai and Professor Y. Narhari of IISc, Bangalore. They especially thank Professor T. J. Sawik, University of Mining and Metallurgy, Krakow, and Professor H. C. Zhang, Texas Tech University.

Appendix

In sequence to illustrate the proposed solution methodology, consider the example problem given in table 1. The solution steps are as follows.

Step 1. $J = 8, M = 4, T_m = 480, V_m = 1\text{-}4.$
$S_m = 5, V_m = 1\text{-}4, q_1 = 8, q_2 = 9, q_3 = 13,$
$q_4 = 6, q_5 = 9, q_6 = 10, q_7 = 12$ and $q_8 = 13.$

The remaining data can be referred from table 1.

Step 2. $T_1 = T_2 = T_3 = T_4 = 480$ min.
$S_1 = S_2 = S_3 = S_4 = 5$ numbers.
$(ET'_1)_{t=0} = 312$ min, $(ES'_1)_{t=0} = 3$
$(ET'_2)_{t=0} = 423$ min, $(ES'_2)_{t=0} = 2$
$(ET'_3)_{t=0} = 371$ min, $(ES'_3)_{t=0} = 5$
$(ET'_4)_{t=0} = 766$ min, $(ES'_4)_{t=0} = 6.$

Steps 3 and 4.

Job no.	q_1	$\sum_{k=1}^{K} te_{kjm}$	$\sum_{k=1}^{K} to_{kjm}$	μ_q^j	μ_e^j	μ_o^j
1	8	144	0	0.285	0.996	0
2	9	414	225	0.428	0	0.540
3	13	143	338	1	1	0.812
4	6	198	0	0	0.797	0
5	9	225	198	0.428	0.697	0.475
6	10	160	280	0.571	0.937	0.384
7	12	276	384	0.857	0.509	0.663
8	13	312	416	1	0.376	0.750

$(q_j)_{\max} = 13, (q_j)_{\min} = 6.$

$$\left(\sum_{k=1}^{K} te_{kjm}\right) \max = 414, \left(\sum_{k=1}^{K} te_{kjm}\right) \min = 143$$

$$\left(\sum_{k=1}^{K} to_{kjm}\right) \max = 416, \left(\sum_{k=1}^{K} to_{kjm}\right) \min = 0.$$

Step 5. $\mu_s^1 = 0.427, \mu_s^4 = 0.265, \mu_s^7 = 0.676,$
$\mu_s^2 = 0.322, \mu_s^5 = 0.533, \mu_s^8 = 0.708,$
$\mu_s^3 = 0.937, \mu_s^6 = 0.630,$

Assume $w_e = w_o = w_q = 1$.

$S - \{3,8,7,6,5,1,2,4\}$.

Step 6. Allocate job 3
Step 6.1. First job in sequence is job 3.
Step 6.2. $\mathrm{OMAV}_1^3 = \mathrm{M1}(338,2) + \mathrm{M3}(143,3)$
$\mathrm{OMAV}_2^3 = \mathrm{M4}(338,2) + \mathrm{M3}(143,3)$.
Step 6.3.

	(T_m, S_m)	(AT_m, AS_m)	(ET'_m, ES'_m)	(ET_m, ES_m)	$(RT_m[\mathrm{OMAV}_1^3], RS_m[\mathrm{OMAV}_1^3])$	$(RT_m[\mathrm{OMAV}_2^3], RS_m[\mathrm{OMAV}_2^3]$
M1	480, 5	480, 5	312, 3	312, 3	142, 3	312, 3
M2	480, 5	480, 5	423, 2	423, 2	480, 5	423, 2
M3	480, 5	480, 5	371, 5	228, 2	337, 2	228, 2
M4	480, 5	480, 5	766, 6	766, 6	142, 3	766, 6

Steps 6.4.

	$\frac{RT_m[\mathrm{OMAV}_1^3] - Et_m}{AT_m - ET}$	$\frac{RS_m[\mathrm{OMAV}_1^3] - Es_m}{AS_m - ES_m}$	$\frac{RT_m[\mathrm{OMAV}_2^3] - Et_m}{AT_m - ET_m}$	$\frac{RS_m[\mathrm{OMAV}_2^3] - ES_m}{AS_m - ES_m}$
M1	$\frac{142-312}{480-312}(=-1.01)$	$\frac{3-3}{5-3}(=0)$	$\frac{480-312}{480-312}(=1)$	$\frac{5-3}{5-3}=(1)$
M2	1	1	1	1
M3	0.432	0	0.432	0
M4	1	1	2.18	3

$$\max\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_1^3]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)=1$$

$$\min\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_1^3]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)=-1.01$$

$$\max\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_1^3]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)=1$$

$$\min\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_1^3]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)=0$$

$$\max\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_2^3]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)=2.18$$

$$\min\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_2^3]-\mathrm{ET}_m}{\mathrm{AT}_m-\mathrm{ET}_m}\right)=0.432$$

$$\max\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_2^3]-\mathrm{ET}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)=3$$

$$\min\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_2^3]-\mathrm{ES}_m}{\mathrm{AS}_m-\mathrm{ES}_m}\right)=0$$

$$\begin{aligned}
[\mathrm{OMAV}_1^3] &= \frac{1}{4}\left\{\frac{(-1.01)-(-1.01)}{1-(-1.01)}+\frac{1-(-1.01)}{1-(-1.01)}+\frac{0.432-(-1.01)}{1-(-1.01)}\right.\\
&\quad\left.+\frac{1-(-1.01)}{1-(-1.01)}\right\}\times\frac{1}{4}\left\{\frac{0-0}{1-0}+\frac{1-0}{1-0}+\frac{0-0}{1-0}+\frac{1-0}{1-0}\right\}\\
&= \tfrac{1}{4}\{0+1+0.712+1\}\times\tfrac{1}{4}\{0+1+0+1\}\\
&= 0.678\times 0.5\\
&= 0.339.
\end{aligned}$$

Similarly, $\mu[\mathrm{OMAV}_2^3]=0.170$
Since $\mu[\mathrm{OMAV}_1^3]>\mu[\mathrm{OMAV}_2^3]$, OMAV_1^3 is selected.

Step 6.6. Updating:

	AT_m	AS_m	ET'_m	ES'_m
M1	142	3	312	3
M2	480	5	423	2
M3	337	2	228	2
M4	480	5	766	6

Step 6.7. Allocate next job in the sequence, i.e. job 8.

Step 6.2.

$$\mathrm{OMAV}_1^8 = \mathrm{M}1(325,1) + \mathrm{M}1(91,1) + \mathrm{M}1(312,3)$$

$$\mathrm{OMAV}_2^8 = \mathrm{M}1(325,1) + \mathrm{M}1(91,1) + \mathrm{M}1(312,3)$$

$$\mathrm{OMAV}_3^8 = \mathrm{M}1(325,1) + \mathrm{M}1(91,1) + \mathrm{M}1(312,3)$$

$$\mathrm{OMAV}_4^8 = \mathrm{M}1(325,1) + \mathrm{M}1(91,1) + \mathrm{M}1(312,3)$$

$$\mathrm{OMAV}_5^8 = \mathrm{M}1(325,1) + \mathrm{M}1(91,1) + \mathrm{M}1(312,3)$$

$$\mathrm{OMAV}_6^8 = \mathrm{M}1(325,1) + \mathrm{M}1(91,1) + \mathrm{M}1(312,3).$$

Step 6.3.

	ET_m, ES_m		$\mathrm{RT}_m[\mathrm{OMAV}_1^8]$, $\mathrm{RT}_s[\mathrm{OMAV}_1^8]$,	$\mathrm{RT}_m[\mathrm{OMAV}_2^8]$, $\mathrm{RT}_s[\mathrm{OMAV}_2^8]$,	$\mathrm{RT}_m[\mathrm{OMAV}_3^8]$, $\mathrm{RT}_s[\mathrm{OMAV}_3^8]$,	$\mathrm{RT}_m[\mathrm{OMAV}_4^8]$, $\mathrm{RT}_s[\mathrm{OMAV}_4^8]$,	$\mathrm{RT}_m[\mathrm{OMAV}_5^8]$, $\mathrm{RT}_s[\mathrm{OMAV}_5^8]$,	$\mathrm{RT}_m[\mathrm{OMAV}_6^8]$, $\mathrm{RT}_s[\mathrm{OMAV}_6^8]$,
M1	0	0	−586, −2	−261, −1	−261, −1	−495, −1	−170, 0	−170, 0
M2	423	2	480, 5	155, 4	480, 5	389, 4	64, 3	389, 4
M3	228	2	337, 2	337, 2	12, 1	337, 2	337, 2	12, 1
M4	776	6	480, 5	480, 5	480, 5	480, 5	480, 5	480, 5

Step 6.4. $\mu[\mathrm{OMAV}_1^8] = \mu[\mathrm{OMAV}_2^8] = \mu[\mathrm{OMAV}_3^8] = \mu[\mathrm{OMAV}_4^8] = 0$

$\mu[\mathrm{OMAV}_5^8] = 0.391, \mu[\mathrm{OMAV}_6^8] = 0.286.$

$\mu[\mathrm{OMAV}_6^8]$ is given by,

$$\max\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_6^8] - \mathrm{ET}_m}{\mathrm{AT}_m - \mathrm{ET}_m}\right) = 1$$

$$\min\left(\frac{\mathrm{RT}_m[\mathrm{OMAV}_6^8] - \mathrm{ET}_m}{\mathrm{AT}_m - \mathrm{ET}_m}\right) = -1.98$$

$$\min\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_6^8] - \mathrm{ES}_m}{\mathrm{AS}_m - \mathrm{ES}_m}\right) = \frac{1-2}{2-2} = 0$$

$$\max\left(\frac{\mathrm{RS}_m[\mathrm{OMAV}_6^8] - \mathrm{ES}_m}{\mathrm{AS}_m - \mathrm{ES}_m}\right) = 1$$

as $\mathrm{RS}_m[\mathrm{OMAV}_6^8] \neq \mathrm{AS}_m$ and $\mathrm{AS}_m = \mathrm{ES}_m$.

Hence,

$$\mu[\mathrm{OMAV}_6^8] = \tfrac{1}{4}\,\{0.262 + 0.464 + 0 + 1\}\,\tfrac{1}{4}\,\{0 + 0.66+) + 1\}$$
$$= 0.431\ 0.665 = 0.286$$

Step 6.5. OMAV_5^8 is selected.
Step 6.6. updation:

	AT_m	AS_m	ET'_m	ES'_m
M1	−170	0	0	0
M2	64	3	423	2
M3	337	2	228	2
M4	480	5	776	6

Step 6.7. Allocate next job in the sequence, i.e. job 7.
Step 6.2.

$$\mathrm{OMAV}_1^7 = \mathrm{M2}(228,1) + \mathrm{M1}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_2^7 = \mathrm{M2}(228,1) + \mathrm{M1}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_3^7 = \mathrm{M2}(228,1) + \mathrm{M1}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_4^7 = \mathrm{M3}(228,1) + (\mathrm{M1}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_5^7 = \mathrm{M3}(228,1) + \mathrm{M2}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_6^7 = \mathrm{M3}(228,1) + \mathrm{M3}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_7^7 = \mathrm{M4}(228,1) + \mathrm{M1}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_8^7 = \mathrm{M4}(228,1) + \mathrm{M2}(156,1) + \mathrm{M4}(276,3)$$

$$\mathrm{OMAV}_9^7 = \mathrm{M4}(228,1) + \mathrm{M3}(156,1) + \mathrm{M4}(276,3).$$

Note. Steps 6.3 and 6.4 involve repetative calculation and hence the results are:

$$\mu[\mathrm{OMAV}_1^7] = \mu[\mathrm{OMAV}_4^7] = \mu[\mathrm{OMAV}_7^7] = 0$$

$$\text{as}\, \mathrm{RS}_1[\mathrm{OMAV}_1^7],$$

$$\mathrm{RS}_1[\mathrm{OMAV}_4^7],$$

$$\mathrm{RS}_1[\mathrm{OMAV}_7^7] < 0.$$

$$\mu[\mathrm{OMAV}_8^7] = 0.1586 > \mu[\mathrm{OMAV}_2^7],\ \mu[\mathrm{OMAV}_3^7],$$

$$\mu[\mathrm{OMAV}_5^7],\ \mu[\mathrm{OMAV}_6^7],\ \mu[\mathrm{OMAV}_9^7].$$

Step 6.5. OMAV_8^7 is selected.
Step 6.6. updation:

	AT_m	AS_m	ET'_m	ES'_m
M1	−170	0	0	0
M2	−92	2	423	2
M3	337	2	228	2
M4	−24	1	500	3

Step 6.7. Allocate next job in the sequence, i.e. job 6.

Step 6.2.

$$\mathrm{OMAV}_1^6 = \mathrm{M4}(160,1) + \mathrm{M2}(70,1) + \mathrm{M1}(210,1)$$

$$\mathrm{OMAV}_2^6 = \mathrm{M4}(160,1) + \mathrm{M3}(70,1) + \mathrm{M1}(210,1)$$

$$\mathrm{OMAV}_3^6 = \mathrm{M4}(160,1) + \mathrm{M4}(70,1) + \mathrm{M1}(210,1)$$

$$\mathrm{OMAV}_4^6 = \mathrm{M4}(160,1) + \mathrm{M2}(70,1) + \mathrm{M2}(210,1)$$

$$\mathrm{OMAV}_5^6 = \mathrm{M4}(160,1) + \mathrm{M3}(70,1) + \mathrm{M2}(210,1)$$

$$\mathrm{OMAV}_6^6 = \mathrm{M4}(160,1) + \mathrm{M4}(70,1) + \mathrm{M2}(210,1).$$

Note. Steps 6.3 and 6.4. involve repetative calculation and hence the results are summarized as:

$$\mu[\mathrm{OMAV}_1^6] = \mu[\mathrm{OMAV}_2^6] = \mu[\mathrm{OMAV}_3^6]$$
$$= \mu[\mathrm{OMAV}_6^6] = 0.$$
$$\mu[\mathrm{OMAV}_4^6] > \mu[O\mathrm{MAV}_5^6].$$

Step 6.5. OMAV_4^6 is selected.
Step 6.6. Updation:

	AT_m	AS_m	ET_m'	ES_m'
M1	−170	0	0	0
M2	−372	0	423	2
M3	337	2	228	2
M4	−184	0	340	2

Step 6.7. Allocate next job in the sequence, i.e. job 5.
Step 6.2.

$$\mathrm{OMAV}_1^5 = \mathrm{M2}(198,2) + \mathrm{M2}(225,1)$$

$$\mathrm{OMAV}_2^5 = \mathrm{M3}(198,2) + \mathrm{M2}(225,1).$$

Note. Steps 6.3 and 6.4. involve repetative calculation and hence the reuslts are summarized as:

$$\mu[\mathrm{OMAV}_1^5] = \mu[\mathrm{OMAV}_2^5] = 0.$$

$$\text{as } \mathrm{RS}_m[\mathrm{OMAV}_1^5],\ \mathrm{RS}_m[\mathrm{OMAV}_2^5] < 0, \text{ for } m = 2.$$

Step 6.5. Job 5 is rejected.
Step 6.6. As job 5 is rejected, the status is already updated.
Step 6.7. Allocate next job in the sequence, i.e. job 1.
Step 6.2. $\mathrm{OMAV}_1^1 = \mathrm{M3}(144,1)$.
Steps 6.3–5: OMAV_1^1 is selected.
Step 6.6. Updation:

	AT_m	AS_m	ET'_m	ES'_m
M1	−170	0	0	0
M2	−372	0	423	2
M3	193	1	84	1
M4	−184	0	340	2

Step. 6.7. Allocate next job in the sequence, i.e. job 2.
Step 6.2.

$$\mathrm{OMAV}_1^2 = \mathrm{M1}(225,1) + \mathrm{M4}(216,2) + \mathrm{M2}(198,1)$$

$$\mathrm{OMAV}_2^2 = \mathrm{M4}(225,1) + \mathrm{M4}(216,2) + \mathrm{M2}(198,1).$$

Steps 6.3–6. Job 2 is rejected.
Step 6.6. Updation: already updated.
Step 6.7. Allocate next job in the sequence, i.e. job 4.
Step 6.2.

$$\mathrm{OMAV}_1^4 = \mathrm{M3}(84,1) + \mathrm{M4}(114,1).$$

Steps 6.3–5. Job 4 is rejected.

Step 7.

$$\mathrm{U} = \{5,2,4\}$$

$$\mathrm{A} = \{3,8,7,6,1\}$$

$$\mathrm{SU_A} = \sum_{m=1}^{M} \mathrm{RT}_m = [(-170) + (-372) + 193 + (-184)]$$
$$= -533.$$

$$\mathrm{Th_A} = \sum_{j=1, J \in A}^{J} q_\mathrm{A} = [13 + 13 + 12 + 10 + 8] = 56.$$

Step 8. $\mathrm{SU_A}$ is negative and go for reallocation.
Step 9.1.

$$\mathrm{SU_{A8}} = -533 + 728 = 195.$$

$$\mathrm{SU_{A7}} = -533 + 660 = 127.$$

Rejection of job 7 from the set of assigned job gives system unbalance of 127 and throughput of 44 (i.e. minimum positive).

Step 9.2. Updation:

	AT_m	AS_m	ET'_m	ES'_m
M1	−170	0	0	0
M2	−216	1	423	2
M3	193	1	84	1
M4	320	4	616	5

$$A = \{3,8,6,1\}$$

$$U = \{5,2,4,7\}.$$

Step 9.3. Allocation of job 5 from set U: job 5 cannot be allocated due to tool slot constraints on machines 2 and 3.
Allocate job 2 from Set U:
$OMAV_1^2$ cannot be selected due to tool slot constraint on M1.
$OMAV_2^2$ is hence selected. Updation yields:

	AT_m	AS_m	ET'_m	ES'_m
M1	−170	0	0	0
M2	−414	0	225	1
M3	193	1	84	1
M4	−121	2	175	3

$SU_A = -512, Th_A = 53$
$A = \{3,8,6,1,2\}$ $U = \{5,4,7\}$.
Allocate job 4 from set U:

	AT_m	AS_m	ET'_m	ES'_m
M1	−170	0	0	0
M2	−216	1	423	2
M3	109	0	0	0
M4	206	3	502	4

$SU_A = -71, Th_A = 50$
$A = \{3,8,6,1,4\}$ $U = \{5,2,7\}$
No improvement.

Step 9.4. $SU_A = 127, Th_A = 44$
$A = \{3,8,6,1\}$
And $SU = 127, Th = 44$.

References

Ammons, J. C., Lofgren, C. B. and McGinnis, 1985, A large scale machine-loading problem in flexible assembly. *Annals of Operation Research*, **3**, 319–322.

Bellman, R. E. and Zadeh, L. A., 1970, Decision-making in a fuzzy environment. *Management Science*, **17**, B141–164.

Berrada, M. and Stecke, K. E, 1986, A Branch and bound approach for machine load balancing in flexible manufacturing systems. *Management Science*, **32**, 1316–1335.

Chen, Y. J. and Askin, R. G., 1990, A multiobjective evaluation of flexible manufacturing system loading heuristics. *International Journal of Production Research*, **25**, 895–911.

Choi, R. H. and Malstorm, E. M., 1988, Evaluation of traditional work scheduling rules in a FMS with a physical simulator. *Journal of Manufacturing System*, **7**, 3032–3045.

Dallery, Y. and Stecke, K. E., 1990, On the optimal allocation of servers and workloads in closed queuing networks. *Operations Research*, **37**, 694–703.

DUBOIS, D. and PRADE, H., 1980, *Fuzzy Sets and Systems: Theory and Application* (New York: Academic Press).

ESCUDERO, L. F., 1989, An exact algorithm for part input sequencing and scheduling with wide constraints in FMS. *International Journal of Flexible Manufacturing System*, **1**, 143–174.

GYMPAH, K. A. and MEREDITH, J. R., 1996, A simulation study of FMS tool allocation procedures. *Journal of Manufacturing System*, **15**, 419–431.

HWANG, S. S., 1986, Models for production planning in flexible manufacturing system. PhD thesis, University of California, Berkeley.

JAIN, S., BARKER, K. and SOTERFLED, D., 1989, Expert simulation for on-line scheduling. *Proceedings of the 1989 Winter Simulation Conference*, pp. 930—935.

KAZEROONI, A., CHAN, F. T. S. and ABHARY, K., 1997, Real-time operation selection in FMS using simulation — a fuzzy approach. *Production Planning and Control*, **8**, 771–779.

KIM, M. M. and KIM, Y.-D., 1994, Simulation based real time scheduling in a FMS. *Journal of Manufacturing System*, **13**, 85–93.

KIM, Y. D. and YANO, C. A., 1993, A heuristic approach for loading problems in flexible manufacturing system. *IIE Transactions*, **25**, 26–39.

KIM, Y. D. and YANO, C. A., 1994, A new branch-and-bound algorithm for loading problems in flexible manufacturing system. *International Journal of Flexible Manufacturing System*, **6**, 361–382.

KIM, Y. D. and YANO, C. A., 1997, Impact of throughput based objectives and machine grouping decisions on the short-term performance of flexible manufacturing systems. *International Journal of Production Research*, **35**, 3303–3322.

KUMAR, P., TIWARI, N. K. and SINGH, N., 1990, Joint Consideration of grouping and loading problem in a flexible manufacturing system. *International Journal of Production Research*, **28**, 1345–1356.

LIANG, M. and DUTTA, S. P., 1992, Combined part-selection, load-sharing and machine-loading problem in flexible manufacturing system. *International Journal of Production Research*, **13**, 2335–2349.

LIANG, M. and DUTTA, S. P., 1993a, Solving a combined part-selection, machine loading and tool-configuration problem in flexible manufacturing system. *Production and Operations Management*, **2**, 97–113.

LIANG, M. and DUTTA, S. P., 1993b, An Integrated approach to part-selection and machine-loading problem in a class of flexible manufacturing systems. *European Journal of Operational Research*, **67**, 387–404.

MORENO, A. A. and DING, F.-Y., 1993, Heuristic for FMS loading and part-type selection problems. *International Journal of Flexible Manufacturing System*, **5**, 287–300.

MUKHOPADHYAY, S. K., 1999, Machine loading of FMS measuring fuzzy entropy. *Proceedings of the 15th ICPR*, Ireland, pp. 505–508.

MUKHOPADHYAY, S. K., MAITI, B. and GARG, S., 1991, Heuristic solution to the scheduling problem in flexible manufacturing system. *International Journal of Production Research*, **29**, 2003–2024.

MUKHOPADHYAY, S. K., MIDHA, S. and KRISHNA, V. A., 1992, A heuristic procedure for loading problem in flexible manufacturing system. *International Journal of Production Research*, **30**, 2213–2228.

MUKHOPADHYAY, S. K., SINGH, M. K. and SRIVASTAVA, R., 1998, FMS machine loading: a simulated annealing approach. *International Journal of Production Research*, **36**, 1526–1547.

MUKHOPADHYAY, S. K. and TIWARI, M. K., 1995, Solving machine-loading problem of FMS by conjoint measurement. *Proceedings of the 13th ICPR*, Jerusalem, pp. 74–76.

NAYAK, G. K. and ACHARYA, D., 1998, Part type selection and machine loading and part type volume determination problem in FMS planning. *International Journal of Production Research*, **36**, 1801–1824.

NOTO LA DIEGA, S., PASSANNATI, A. and PERRONE, G., 1995, Machining economics in FMS by a fuzzy approach. *Annals of the CIRP*, **44**, 417–420.

O'GRADY, P. J. and MENON, U., 1987, Loading a flexible manufacturing system. *International Journal of Production Research*, **25**, 1053–1068.

ROSS, T. J., 1997, Fuzzy Logic with Engineering Applications (Singapore: McGraw-Hill).

SARIN, S. C. and CHEN, C. S., 1987, The machine loading and tool allocation problem in a flexible manufacturing system. *International Journal of Production Research*, 25, 1081–1094.

SAWIK, T. J., 1988, Modelling and scheduling a batch-type production on identical machines. *European Journal of Operational Research*, **35**, 393–400.

SAWIK, T. J., 1990, Modelling and scheduling of a flexible manufacturing system. *European Journal of Operational Research*, **45**, 177–190.

SAWIK, T. J., 1996, A multilevel machine and vehicle scheduling in a flexible manufacturing system. *Mathematical and Computer Modelling*, **23**, 45–57.

SAWIK, T. J., 1997, An interactive approach to bicriterion loading of a flexible assembly system. *Mathematical and Computer Modelling*, **25**, 71–83.

SAWIK, T. J., 1998, A lexicographic approach to bi-objective loading of a flexible assembly system. *European Journal of Operational Research*, **107**, 656–668.

SHANKER, K. and SRINIVASULU, A., 1989, Some solution methodologies for a loading problem in a flexible manufacturing system. *International Journal of Production Research*, **27**, 1019–1034.

SHANKER, K. and TZEN, Y. J., 1985, A loading and despatching problem in a random FMS. International Journal of Production Research, **23**, 579–595.

SHANTHIKUMAR, J. J. and YAO, D. D., 1987, Optimal server allocation in a system of multi-server stations. *Management Science*, **33**, 1173–1182.

SHANTHIKUMAR, J. J. and YAO, D. D., 1988, On Server allocation in multiple centre manufacturing system. *Operation Research*, **36**, 333–342.

SINGH, N. and MOHANTY, B. K., 1991, A fuzzy approach to multi-objective routing problem with application to process planning in a manufacturing system. *International Journal of Production Research*, **23**, 1161–1170.

SINGHAL, K., FINE, C. H., MEREDITH, J. R. and SURI, R., 1987, Research and Models for automated manufacturing. *Interfaces*, **17**, 5–14.

STECKE, K. E., 1983, Formulation and Solution of nonlinear integer production planning problem for flexible manufacturing system. *Management Science*, **29**, 273–288.

STECKE, K. E. and MORIN, T. L., 1985, The optimality of balancing workload in certain types of flexible manufacturing system. *European Journal of Operational Research*, **20**, 68–82.

STECKE, K. E. and SOLBERG, J. J., 1981, Loading and control policies for a flexible manufacturing system. *International Journal of Production Research*, **19**, 481–490.

TIWARI, M. K., HAZARIKA, B., VIDYARTHI, N. K., JAGGI, P. and MUKHOPADHYAY, S. K., 1997, A heuristic solution approach to the machine-loading problem of FMS and its Petrinet model. *International Journal of Production Research*, **35**, 2269–2284.

YU, L., SHIH, H. M. and SEKIGUCHI, T., 1999, Fuzzy inference-based multiple criteria FMS scheduling. *International Journal of Production Research*, **37**, 2315–2333.

ZADEH, L. A., 1965, Fuzzy sets. *Information and Control*, **8**, 338–353.

ZADEH, L. A., 1991, Fuzzy logic principles, applications and perspectives [Public lecture], University of Oklahoma, Norman, 18 April.

ZHANG, H.-C. and HUANG, S. H., 1994, A fuzzy approach to process plan selection. International Journal of Production Research, **32**, 1265–1279.

ZIMMERMAN, H.-J., 1991, *Fuzzy Set Theory and its Application* (Boston: Kluwer-Nijhoff.)

ZIMMERMAN, H. J., 1992, Approximate reasoning in manufacturing. In A. Kusiah (ed.), *Intelligent Design and Manufacturing* (Chichester: Wiley), pp. 701–722.