Babol Noshirvani University of Technology

10th Iranian Conference on Manufacturing Engineering
ICME 2010
March 1st - 3rd 2010
Babol Noshirvani University of Technology

Society of Manufacturing Engineering of Iran

A LINEAR MATHEMATICAL PROGRAMMING MODEL FOR MACHINE LOADING PROBLEM IN A FLEXIBLE MANUFACTURING SYSTEM

Maghsud Solimanpur [1], Amir Musa Abazari *, [2] and Hossein Sattari [3]

1. Associate Professor, Faculty of Engineering, Urmia University
2. Master of Science Student, Faculty of Engineering, Urmia University
3. Master of Science Student, Faculty of Engineering, Urmia University

Abstract

The evolution of flexible manufacturing systems (FMSs) offers great potential for increasing flexibility by ensuring both cost effectiveness and customized manufacturing at the same time. Machine loading problem of a FMS encompasses various types of flexibility aspects pertaining to part selection and operation assignments. This paper proposes a linear mathematical programming model for job selection and operation allocation problem of FMS to maximize profitability and utilization of system. The proposed model assigns operations to different machines considering the capacity of tool magazines, capacity of machines, batch sizes, tool slots, etc. The attempted model has been coded in Lingo Software to demonstrate applicability of the proposed model to industrial cases. Performance of the proposed model is evaluated based on some benchmark problems adopted from the literature. Comparison of the results with those published in the literature indicates supremacy of the proposed model.

Keywords: Flexible manufacturing systems - Machine loading problem - Operation allocating

1. Introduction

A flexible manufacturing system (FMS) can be defined as an integrated configuration of numerical control (NC) machine tools, other auxiliary production equipment, and a material handling system designed to simultaneously manufacture a low to medium volumes of a wide variety of high quality products at low cost [1]. In a FMS, we can have benefits of both flow shop and job shop factories, though we may face complexities for installing such a system. i.e. production management of an FMS is more difficult than that of production lines or job shops because of the additional flexibility-related degrees of freedom.

The benefits that can be accrued due to installation of an FMS are: increased machine utilization, fewer machines, reduction in required factory floor space, greater responsiveness to changes, reduced inventory requirements, lower manufacturing lead times, reduced direct labor requirement, higher labor productivity, opportunity for automated production, etc. [2].

In general, FMS operational decisions consist of pre- and post-release decisions. Pre-release decisions, also called the FMS planning problem, consider the pre-arrangement of parts and tools before the FMS begins to process. Post-release decisions, also called the FMS scheduling problem, deal with sequencing and routing of parts, when the system is in operation. Pre-release decisions, viz. machine grouping, part type selection, production ratio determination, resource allocation and

loading problems must be solved while setting up of a FMS. Among pre-release decisions, machine loading is considered as one of the most vital production planning problems because performance of FMS largely depends on it. Loading problem, in particular, deals with allocation of jobs to various machines under technological constraints with the objective of meeting certain performance measures.
Stecke [3], Sarin and Chen [4] divided machine loading problem into five subproblems:

1. Machine grouping;
2. Part type selection;
3. Production rate determination;
4. Resource allocation;
5. Loading.

Having machine grouping, part type selection, production ratio and resource allocation problems solved, the loading problem is specified as selecting a subset of jobs from a job pool and assigning their operations to the appropriate machines in the forthcoming planning period so as to achieve certain specified objectives while meeting the system constraints. Stecke has described six objectives in loading a FMS [3]:

1. Balancing the machine processing time;
2. Minimizing the number of movements;
3. Balancing the workload per machine for a system of groups of pooled machines of equal sizes;
4. Unbalancing the workload per machine for a system of pooled machines of unequal sizes;
5. Filling the tool magazines as densely as possible;
6. Maximizing the sum of priorities of operations.

It is clear from above that the machine loading problem involves multiple objectives. Ammons et al. [5] resolve the loading problem considering a bi-criteria objective of balancing workload and minimizing work stations visits whereas Shankar and Tzen [6] consider balancing workload and meeting due-date of part types. Tiwari et al. [7] and Mukhopadhyay et al. [8] tackle machine loading problem using heuristic approaches with an objective of minimizing system unbalance and maximizing throughput. Swarnkar et al. [10] develop a systematic integrated procedure to address constituents of machine loading problem simultaneously and use a hybrid tabu search and simulated annealing based heuristic approach for solving it. They consider minimization of system unbalance and maximization of throughput as their objective function by taking into account the technological constraints such as available machining time and tool slots on machines. Prakash et al. [9] used a modified immune algorithm for machine loading problem with the objectives same as Swarnkar et al. [10] and considered both underutilized and overutilized times on machines for calculating system unbalance.
In general, machine loading problems can be addressed mainly by two approaches: (a) heuristic oriented and (b) optimization based methods. Heuristic approaches are largely based upon rules and rely on empirical experiences. Therefore, one of the limitations of a heuristic approach is in its difficulty to estimate results in a new or

completely changed environment. While optimization based methods such as-integer programming, dynamic programming, branch and bound, etc. are robust in applicability, they tend to become impractical when problem size increases.
In summary, the machine loading problems pertaining to automated manufacturing systems belong to the category of NP-hard problems. To illustrate the complexity associated with machine loading problems, assume the data shown in Table 1.

Table 1: Description of problem no.1 (adopted from Mukhopadhayay et al. [8])

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

There exist 8 jobs, which can be sequenced in 8! ways, all together 2592 combination of operation–machine allocation are possible for one of the job sequences. Hence, for 8! job sequences the total number of possible allocation turns out to be 2592×8! = 104509440. Some of these allocations are not possible because they are not able to satisfy system constraints such as available machining time and tool slots. Enumerating an optimal/near optimal solution in such a huge search space is computationally complex.
This paper 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, and the product may require several operations and may have

alternative processing routes i.e. several types of machines may be capable of processing the same operation and the system may have more than one machine of the same type.
In this article, machine loading problem can be viewed as selecting a subset of jobs from the job pool and allocating them among machines. The objective considered is minimization of system unbalance and maximization of profitability while satisfying the constraints, viz. available machining time and tool slots. Furthermore, the unit profit of jobs is assumed to be 1 in this paper to make the results of the proposed model comparable with those reported for other methods in the literature. The proposed model has been tested on ten problems and the results are compared with solutions obtained by Modified Immune Algorithm reported in Prakash et al. [9]. Results show that the proposed model is very reliable and efficient.
The paper is organized as follows. In section 2, problem environment is described and modeled. In section 3, the results of proposed model are compared with the Modified Immune Algorithm (MIA) [9]. The conclusions are given in section 4.

2. Experimental Procedures

2.1. Problem description

The FMS under consideration in this paper consists of a number of multifunctional CNC machines and tools with the potential to execute several operations, automated material handling devices and other amenities, where several types of jobs arrive with varied processing requirements. Jobs are available in batches and some of them are to be selected for processing during a given planning horizon. Job selection and loading constitute two major components of a tactical planning problem of any FMS. The job selection problems are concerned with selecting the sets of jobs to be produced during the upcoming planning horizon while the loading problem involves allocation of operations and required tools for the selected jobs on the machines.
A job includes one or more operations and each of them can be performed by one or more machines. The particularities related to the production requirement of the job, number of operations for each job and their machining time and number of tool slots required by each operation of each job are known in advance. Essential and optional types of operations are allied with each job. Essential operations of a job means that this operation can be performed only on a particular machine using a certain number of tool slots whereas optional operations imply that they can be carried out on a number of machines with the same or varying processing time and tool slots. In this problem, the flexibility lies in the selection of a machine for processing the optional operations of the jobs.
The machine loading problem can be defined as "…given a set of jobs to be produced, set of tools that are needed for processing the jobs on a set of machines, and using a set of resources such as material handling systems, pallets and fixtures, how the job be assigned and tools be allocated so that some measures of productivity is optimized".
The complexity associated with machine loading problem has been discussed in the previous section. Due to such a large complexity, it is fairly difficult to evaluate the

optimal solution of operation allocations on machines in the presence of several alternatives for the given problem.
In the following subsections, formulation of objective functions and constraints are discussed in detail.

2.2. Units, terminology and symbols

The notations used to demonstrate the proposed model are shown below.
Subscripts
j index of job; $1 \le j \le J$
m index of machine; $1 \le m \le M$
o index of operation; $1 \le o \le O_j$

Parameters
U_m underutilized time on machine m
O_m overutilized time on machine m
T_m time available on machine m
P_j unit profit of job j
b_j batch size of job j
t_{jom} time required by machine m for operation o of job j
S_{jom} tool slot required by machine m for operation o of job j
Decision variables

$$x_{jom} = \begin{cases} 1 & \text{if operation o of job j is assigned to machine m,} \\ 0 & \text{otherwise.} \end{cases}$$

$$y_j = \begin{cases} 1 & \text{if job j is selected,} \\ 0 & \text{Otherwise.} \end{cases}$$

2.3. Mathematical model

The machine loading problem described above is formulated here as a bi-criterion objective function and then the two objectives are combined.

1. The first objective is to maximize the system profitability:

$$\text{Max. } \frac{\sum_{j=1}^{J} P_j b_j Y_j}{\sum_{j=1}^{J} P_j b_j}$$

2. The second objective is to minimize the system unbalance:

$$\text{Min. } \sum_{m=1}^{M} (U_m + O_m) \qquad \text{or} \qquad \text{Min. } \frac{\sum_{m=1}^{M} (U_m + O_m)}{\sum_{m=1}^{M} T_m}$$

This is equivalent to:

$$\text{Max.} \quad -\frac{\sum_{m=1}^{M}(U_m + O_m)}{\sum_{m=1}^{M} T_m}$$

Thus the overall objective function is:

$$\text{Max.} \quad \frac{\sum_{j=1}^{J} P_j b_j Y_j}{\sum_{j=1}^{J} P_j b_j} - \frac{\sum_{m=1}^{M}(U_m + O_m)}{\sum_{m=1}^{M} T_m}$$

Subject to the following constraints:

1. Overloading and under-loading of machines are permitted. This can be expressed as:

$$\sum_{j=1}^{J}\sum_{o=1}^{O_j} b_j . t_{jom} x_{jom} + U_m - O_m = T_m , \qquad \forall m \in 1,...,M$$

2. The number of slots needed for the operations of the jobs to be performed on a machine must always be less than or equal to the tool slots available in that machine. This can be expressed as:

$$\sum_{j=1}^{J}\sum_{o=1}^{O_j} S_{jom} \ x_{jom} \le S_m , \qquad \forall m \in 1,..., M$$

3. Once a job is considered for processing, all the operations are to be completed before undertaking a new job (non-splitting of job), and once a machine is selected for an operation, it has to be completed on the same machine (unique job routing). This can be expressed as:

$$\sum_{m \in B(j,o)} X_{jom} = Y_j , \qquad \forall j \in 1,..., J , \forall o \in 1,..., O_j$$

4. Decision variables are binary. This can be expressed as:

$$x_{jom} = 0, 1 \quad y_j = 0, 1$$

The two different objective functions may be easily attached with weights. However, for the sake of simplicity, we have considered an equal weight to both functions.

3. Results and Discussion

To be comparable with the literature, we assume P_j=1 for all jobs, i.e. the unit profit of all jobs is equal. The attempted model has been coded in Lingo Software. Performance of the proposed model is evaluated based on some benchmark problems adopted from the literature [8]. The proposed model is tested on ten benchmark problems and the results are compared with those reported in Prakash et al. [9]. The solutions for ten problems are given in Table 2 and the comparative results are given in Table 3. From results, it can be seen that the proposed model performs better than the Modified Immune Algorithm (MIA) presented in [9].

Table 2. Summary of results obtained by the proposed model.

Problem number	Jobs assigned	Jobs unassigned	System unbalance	Throughput
1	1,3,5,6,7	2,4,8	228	52
2	1,2,3,4,5,6	-	464	73
3	1,2,3,4,5	-	262	79
4	1,2,3,4,5	-	819	51
5	1,2,3,4,5,6	-	369	76
6	1,2,3,4,5,6	-	336	73
7	1,2,3,4,5,6	-	387	78
8	1,2,5,6,7	3,4	269	54
9	1,2,3,4,5,6,7	-	309	88
10	1,2,3,4,5,6	-	472	67

Table 3. Comparison of proposed model with the MIA [9].

Problem number	Total number of jobs	MIA SU	MIA TH	MIA OFV	Proposed model SU	Proposed model TH	Proposed model OFV
1	8	318	48	0.4344	228	52	0.5313
2	6	524	63	0.5901	464	73	0.7583
3	5	312	69	0.7109	262	79	0.8635
4	5	819	51	0.5734	819	51	0.5734
5	6	536	61	0.5235	369	76	0.8078
6	6	518	61	0.5658	336	73	0.8250
7	6	477	63	0.5593	387	78	0.7984
8	7	677	44	0.2760	269	54	0.6313
9	7	-	-	-	309	88	0.8391
10	6	272	56	0.6942	472	67	0.7542

SU: System Unbalance TH: Throughput
OFV: Objective Function Value

4. Conclusion

The main contribution of this research is to develop an efficient mathematical programming model for solving the machine loading problem for a random FMS. The objective of the loading problem considered in this research is the minimization of system unbalance and maximization of profitability where the system constraints are maximum available time and tool slots on each machine. Some computational experimentation has been carried out to assess the effectiveness of the proposed model. Computational results indicate that the proposed model provides very promising solutions compared with those of Modified Immune Algorithm [9].

It should be pointed out that the proposed solution methodology uses a fixed, predetermined sequencing rather than using a sequencing rule or determining the optimum. To rectify this shortcoming, one can easily solve this problem by using various sequencing rules and then selecting the best solution.
It is worth noting that application of the proposed model is limited to certain cases where there are sufficient number of jigs, fixtures, pallets and automated guided vehicles (AGVs) available in the shop floor. The work may be extended further by imposing constraints on the availability of these resources.

References

1. N.Nagarjuna, O.Mahesh, K.Rajagopal, A heuristic based on multi-stage programming approach for machine loading problem in a flexible manufacturing system, Robotics & Computer-integrated Manufacturing 22 342-352, (2006).
2. G.Mikell P.Automation, Production Systems and Computer Integrated Manufacturing, 2nd ed., India: Pearson, 2003.
3. Stecke KE., Formulation and solution of non-linear integer production planning problem for flexible manufacturing system, Manage Sci 29 273-88, (1983).
4. Sarin S.C., Chen C.S., The machine loading and tool allocation problem in flexible manufacturing system, International Journal of production Research 25 1081-1094, (1987).
5. Ammons J.C., Lofgren C.B., McGinnis L.F., A large scale machine loading problem in flexible assembly, Annals of Operation Research 3 319-322 (1985).
6. Shankar K., Tzen Y.J., A loading and dispatching problem in a random flexible manufacturing system, International Journal of Production Research 16 383-393 (1985).
7. Tiwari M.K., Hazarika B., Jaggi P., Vidyarthi N.K., Mukhopadhyay S.K., A heuristic solution approach to the machine loading problem of an FMS and its petri net model, International Journal of Production Research 35 2269-2284 (1997).
8. Mukhopadhyay S.K., Midha S., Krishna V.M., A heuristic procedure for loading problems in flexible manufacturing systems, International Journal of Production Research 30 2213-2228 (1992).
9. Prakash A., Khilwani N., Tiwari M.K., Cohen Y., Modified Immune Algorithm for job selection and operation allocation problem in flexible manufacturing systems, Advances in Engineering Software 39 219-232 (2008).
10. Rahul Swarnkar, M.K.Tiwari, Modeling machine loading problem of FMSs and its solution methodology using a hybrid tabu search and simulated annealing based heuristic approach, Robotics and Computer Integrated Manufacturing 20 199-209 (2004).