Production Planning of FMS Under Tool Magazine Constraints: A Dynamic Programming Approach

K. Raghava Rau and O. V. Krishnaiah Chetty
Department of Mechanical Engineering, Indian Institute of Technology, Madras, India

Production planning is one of the most important activities for efficient operation of a flexible manufacturing system. This complex activity is concerned with the decisions related to system set-up, involving solving the problems of selection of a set of part types for simultaneous processing, determining the production ratios, assigning the pallets and fixtures, and assigning operations and tools to machines. In this paper, a dynamic programming algorithm is developed to solve the above problems simultaneously by considering the flexibilities and constraints of the system in order to minimise the unbalanced workload of machines. In this context, various factors such as the tools required for operations, alternative routes available for operations, tool magazine capacity and the limited number of pallets and fixtures are considered. The method is validated with a case study.

Keywords: Dynamic programming; Flexible manufacturing systems; Production planning; Tool magazine constraint

1. Introduction

Flexible manufacturing systems (FMSs) are becoming popular because of their ability to manufacture a variety of parts with quick response to market changes. The system flexibility is attributed to the facts that:

1. Each machine is equipped with a tool magazine and an efficient tool exchanger to enhance the system performance.
2. An operation can have alternative machines for processing.

Since an FMS is capital intensive, to justify the cost, it must be used to its optimal performance level.

Operation of an FMS is a complex activity as it involves a large number of decision variables affecting the system performance. Production planning is an important activity in the operation of an FMS as it involves setting up of the system to manufacture a batch of parts. Stecke [1] describes the production planning problems that are to be solved prior to system set-up as part type selection, part mix ratios, allocation of pallets, tool loading and machine grouping. These planning problems are interdependent and unless they are solved simultaneously the solution obtained may not be feasible [2]. For example, one cannot guarantee that all the tools required for the selected part types will fit into the limited slots of the tool magazines, if the problems of part type selection and tool loading are solved independently. Several researchers [1-4] stressed that short-to-medium term production planning must consider the limitations imposed by tool magazine capacity. The limited tool magazine capacity therefore implies that proper part type selection and tool loading procedures must be used to achieve the performance potential of an FMS [5]. Often the part type selection problem is severely constrained by tool loading [6].

Many of the research works have considered only a few aspects of planning and have made restrictive assumptions. The part selection problem has been addressed by Stecke and Kim [7,8] and Lee and Iwata [9]. These papers assume that required tools for the selected parts will fit into the tool magazines of the machines. The tool loading problem has been solved by [10-15]. Tool loading along with scheduling has been solved by [16-25]. All these papers on tool loading have assumed that the parts selection problem is solved a priori.

The purpose of this paper is to solve simultaneously the production planning problems of an FMS, namely, selection of part types, determination of production ratios for the selected part types, assignment of pallets and fixtures, and allocation of operations and tools to machines. In this context, alternative routings for operations with tool requirements, limited pallets and fixtures and capacity constraints of the tool magazines are considered. If tool magazine capacity constraint has to be incorporated in the production planning problem, the formulation of the resulting problem as a mathematical programming model becomes nonlinear. Hence, exact solution methods are not available for solving such models. A dynamic programming (DP) method has been proposed to solve the above problems simultaneously with the objective of minimising the unbalanced workload of machines. The DP method is advantageous compared to other
mathematical programming methods when the complexity of the problem increases.

Section 2 discusses the production planning problem that is to be solved along with the assumptions. Section 3 describes the proposed dynamic programming method for solving the above problem. The method is validated using a published case study, without tool magazine constraints, and the solution to the problem considering tool magazine constraints is discussed in Section 4. The conclusions and scope for future work are given in Section 5. The notation required in using dynamic programming for solving the production planning problem is given at the end of the paper.

2. Problem Statement

Let the system under consideration consist of a set of CNC machines. Each machine has a limited capacity tool magazine. The system can process a number of part types. Each part type has a limited number of dedicated pallets. The following assumptions are made:

1. Tools do not fail and remain with the allocated machines for the planning period.
2. Machines do not fail.
3. There is no refixturing of parts.

The production planning problem is then to determine the set of part types that can be processed in the system simultaneously and their production ratios. In this context, tools required for operations, alternative routes available for operations, tool magazine capacity and the limited number of pallets and fixtures are considered. The objective is to minimise the unbalanced workload of all machines.

3. Dynamic Programming Methodology

Dynamic programming (DP) is a mathematical procedure designed to improve the computational efficiency of selected problems by decomposing them into smaller, and hence computationally simpler, subproblems. It solves the problems in stages through the use of recursive computations using the principle of optimality. Dynamic programming methodology has been applied to different classes of problems like the Knapsack problem [26,27] and the travelling salesman problem [28,29]. The DP method for solving the production planning problem is explained below.

\[ L_{i,k}(S_p(1), n) = n^* p_{w_i,k}, \quad \forall k, \forall p, n=1..n \text{ max and } n \leq f_i \]  
\[ F_i(S_p(1), n, a_i) = \sum_{k=1}^{K} |W_k - L_{i,k}(S_p(1), n)|, \quad \forall p, n=1..n \text{ max} \]  
\[ L_{i,k}(S_p(j), n, a_i) = L_{i,k}(S_p(j), n-a_i) + a_i^* p_{w_i,k} \]  
\[ \forall k, n=1..n \text{ max, } a_i = 1..f_i \text{ and } a_i \leq n-j \]

iff all tools for \( S_p(j) \) can be loaded onto magazines of machine \( k \).

Equations (1) and (2) give the computations involved in the first stage. Equations (3)–(5) define the computations in subsequent stages. Equation (1) calculates the workloads of machines when only one part type is considered at a time, while equation (2) calculates the objective function at the first stage. Equation (3) is a recursive equation for calculating the workloads of all machines for any stage except the first, when \( S_p(j) \) is the set of selected part types and \( n \) is the number of parts in the system and indicates the state in DP. If the tools required for the part types in \( S_p(j) \) subset cannot be loaded then the unbalanced workload for that subset is assigned a very large number, \( B \). Equation (4) is used to calculate the unbalanced workload. In stage \( j \) the number of part types selected is \( j \), i.e. \( |S_p(j)| = j \), where \( j \leq N \). The set \( S(j) \) consists of subsets each of which is the set of part types that can be obtained in stage \( j \). For example, if the available part types are 1,2,3 and 4, then in stage 3 there can be 4 subsets. The subsets are \( \{1,2,3\}, \{1,2,4\}, \{1,3,4\} \) and \( \{2,3,4\} \). The number of subsets to be considered in stage \( j \) is given by \( ^nC_j \). For a given subset of part types the value of a state may vary from \( j \) to \( n \text{ max} \), where \( n \text{ max} \) is a user defined value. For the computations of unbalanced workload at each state for each subset \( S_p(j) \), it is not necessary to continue up to \( n \text{ max} \). Computations are carried out as long as a decreasing trend is observed with \( F^*_j \) values. An increasing trend observed in the \( F^*_j \) value stops the process. The DP calculations can be stopped at the stage where the value of the minimum unbalanced work of that stage is more than the previous stage. Hence the computations required will be very much reduced. The calculations involved in the DP method are detailed below using a numerical example.

An example of a small flexible flow type FMS consisting of 3 part types and 3 machines is considered to explain the proposed dynamic programming method. The processing times of each part type on each machine are given in Table 1. The average workloads for machines M1, M2, and M3 are assumed to be 84, 104, and 104 units of time. In this problem, the calculations are done in 3 stages (i.e. number of part types). Table 2 shows the calculations in the first stage of dynamic programming. In this stage, three subsets of one part type at

Table 1. Processing times of three parts on 3 machines. \( W_1 = 84 \), \( W_2 = 104 \), \( W_3 = 104 \).

<table>
<thead>
<tr>
<th>Part type</th>
<th>M1</th>
<th>M2</th>
<th>M3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11</td>
<td>25</td>
<td>29</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>35</td>
<td>15</td>
<td>5</td>
</tr>
</tbody>
</table>