The International Journal of Flexible Manufacturing Systems, 4 (1991): 5–16

Scheduling Tasks and Vehicles in a Flexible Manufacturing System

JACEK BŁAŻEWICZ
Instytut Informatyki, Politechnika Poznańska, ul. Piotrowo 3a, 60–965, Poznań, Poland

HORST A. EISELT
Faculty of Administration, University of New Brunswick, Fredericton, Canada

GERD FINKE
Laboratoire Artemis, Université de Grenoble, Grenoble, France

GILBERT LAPORTE
Centre de Recherche sur les Transports, Université de Montréal, Montréal, Canada

JAN WEGLARZ
Instytut Informatyki, Politechnika Poznańska, ul. Piotrowo 3a, 60-965, Poznań, Poland

Abstract. Due to their increasing applicability in modern industry, flexible manufacturing systems (FMSs), their design, and their control have been studied extensively in the recent literature. One of the most important issues that has arisen in this context is the FMS scheduling problem. This article is concerned with a new model of an FMS system, motivated by the practical application that takes into account both machine and vehicle scheduling. For the case of a given machine schedule, a simple polynomial-time algorithm is presented that checks the feasibility of a vehicle schedule and constructs it whenever one exists. Then a dynamic programming approach to construct optimal machine and vehicle schedules is proposed. This technique results in a pseudopolynomial-time algorithm for a fixed number of machines.

Key Words: flexible manufacturing systems, production scheduling, vehicle scheduling

1. Introduction

The increasing role of flexible manufacturing systems (FMSs) in modern industry creates the need to analyze their behavior and to work out methods for their design and control. Several questions arise in this context. According to Schmidt (1989) and Stecke (1985), these may be divided into several groups: design problems, planning problems, scheduling problems, and control and monitoring problems. The present article concerns a scheduling problem and is motivated by the practical difficulties associated with the implementation of an FMS by a major North American manufacturer of helicopter parts.

The large number of papers in journals and special volumes on FMSs devoted to the scheduling problem (see, e.g., Kusiak 1986b; Kusiak 1986c; Kusiak and Wilhelm 1989; Schmidt 1989; Stecke and Suri 1985; Stecke and Suri 1988) reflects the importance of this area for the efficient utilization of an FMS. However, as pointed out in Kusiak (1986a), the majority of these papers deal with either part and machine scheduling (Afentakis 1985; Carrie and Petsopoulos 1985; Chang and Sullivan 1984; Erschler et al. 1984; Finke and

Kusiak 1987; Srishkandarajah et al. 1989; Stecke and Solberg 1981) or with automated guided vehicle (AGV) routing (e.g., Villa and Rosetto 1985; Yao 1985). One of the few exceptions is Kusiak (1988) who takes into account tools, fixtures, and pallets as well as machines.

In the present article, two issues, i.e., part and machine scheduling and vehicle routing, are considered together within a framework of FMS scheduling. The objective is to construct a schedule of minimum length that takes into account both problems. The organization of the article is as follows. Section 2 contains the description and modeling of a specific FMS. In section 3, a procedure for vehicle routing is presented, assuming that a production schedule is given. Section 4 deals with simultaneous operation and vehicle scheduling based on a dynamic programming procedure. Some conclusions and possible extensions are given in section 5.

2. Description and modeling of the system

The FMS under consideration has been implemented by a manufacturer (Pratt & Whitney) producing parts for helicopters. A schematic view of the system is presented in Figure 1, and its description is as follows.

Pieces of raw material from which the parts are machined are stored in the automated storage (AS) area (1). Whenever necessary, an appropriate piece of material is taken from storage and moved to the input station (2). This function is performed automatically by computer-controlled pallet changers. Then the piece is transported by an AGV (7) to the desired machine (6), where it is automatically unloaded at the machine's input buffer (8). Every machine in the system is capable of processing any of the required machining operations. This versatility is achieved by having a large number of tools and fixtures that may be used by the machines.

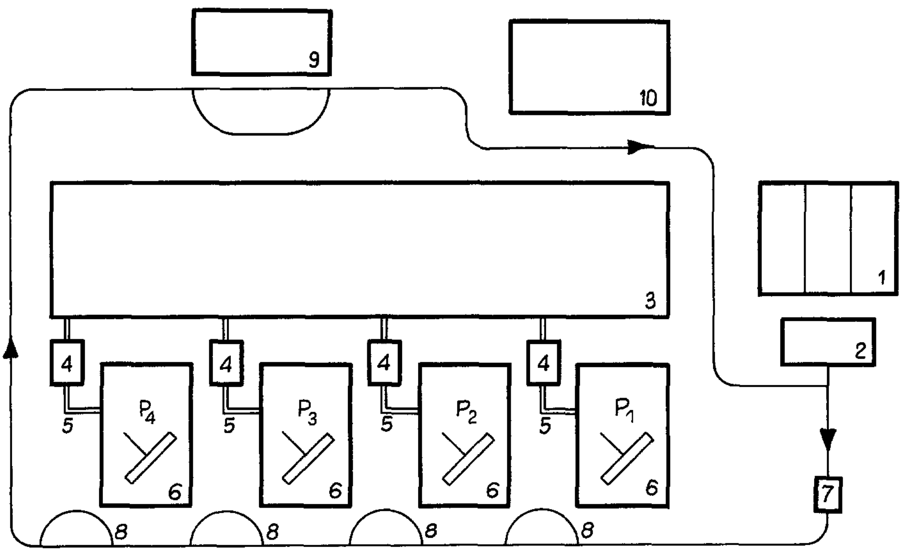

Figure 1. An example FMS system.

Each tool magazine (4) of each machine has a capacity of up to 130 tools, which are used for the various machining operations. The tools of the magazines are arranged in two layers so that the longer tools can occupy two vertical layers. The tools are changed automatically. Fixtures are changed manually. It should be noted that a large variety—almost 100 quite different parts—can be produced by each machine in this particular FMS. Simpler part types require one operation and about 30 tools, and the most complicated parts need about 80 tools. Therefore, the tool magazines have sufficient capacity to stock the tools for one to several consecutive part types in a production schedule. In addition, the tools are loaded from a large, automated central tool-storage area (3), which is located close to the machines. No tool competition is observed, since the storage area contains more than 2000 tools (including many multiple tools) and there are four computer numerically controlled (CNC) machines.

The delivered raw material is mounted manually onto the appropriate fixture and later processed by the tools, which are changed according to a desired plan. The tool technology of this particular system allows the changing of tools during execution of the operations, thereby eliminating the setup times of the tools required for the next operation and occasionally the transfer of a tool to another machine (to validate completely the no-resource competition). The only (negligible) transition time in the FMS that could be observed was in fact the adjustment in size of the spindle that holds the tool whenever the next tool is exchanged with the previous one.

After completion, the finished part exchanges its positions with the raw material of the next operation that is waiting for processing. The part is then automatically transported by an AGV to the inspection section (9). Parts that pass the inspection are transported and unloaded at the storage area (10).

We see that the above system is very versatile due to the usage of many tools and large tool magazines. As pointed out in Jaikumar (1986) and Jaikumar and Van Wassenhove (1987), a common tendency of FMSs is to become so versatile that most of the operations on a part can be accomplished by just one or two machine visits. As a result, many systems consist of identical parallel machines. On the other hand, the existence of a large number of tools in the system allows one not to consider resource (tool) competition. Hence, our problem here reduces in fact to one of simultaneous scheduling and routing of parts among parallel machines. The inspection can be postponed in this analysis, since it is performed separately on a first-come-first-served basis.

Following the above observations, we can model this type of FMS using elements described below. We assume as given a set of n *single-operation part types* $T_1, T_2, \ldots, T_n$ that are to be processed on a set of m parallel identical *machines* $P_1, P_2, \ldots, P_m$, m not being a very large number. (Here parallelism means that every machine is capable of processing any part.) For its processing, every operation T_j requires one arbitrary machine and specified amounts of *additional resources* (Błażewicz et al. 1986), such as *renewable resources*, i.e., different tools used throughout a processing of the operation, and *nonrenewable resources*, which represent raw material. Operation T_j should be granted one machine and the required resources during the noninterrupted *processing time* of length p_j. From the preceding remarks, one may assume that there are enough resources of both categories and that the only scarce resource for which the operations should compete is the machines. Setup times connected with changing tools are assumed to be zero, since tools can be changed on-line during the execution of operations. Setup times resulting from changing part fixtures are included in the processing times.

As mentioned above, machines are identical except for their locations, and thus they require different delivery times. Hence, we may assume that k ($k \leq m$) *automated guided vehicles* (AGVs) $V_1, V_2, \ldots, V_k$ are to deliver pieces of raw material from the storage area to specified machines (or buffers) and the *time* associated with the *delivery* is equal to τ_i, $i = 1, 2, \ldots, m$. The delivery time includes *loading time* at the storage area and *unloading time* at the required machine. During each trip, exactly one piece of raw material is delivered; this is due to the dimension of parts to be machined. After delivery of a piece of raw material, the vehicle takes a pallet with a processed part (possibly from another machine), delivers it to the inspection stage, and returns to the storage area (1). The round trip takes A units of time, including two loading and two unloading times whose sum is equal to a.

It is apparent that the most efficient usage of vehicles in the sense of a throughput rate for parts delivered is achieved when the vehicles are operating in a cyclic mode with cycle time equal to A (because then no idle time in the vehicle schedule exists). In order to avoid traffic congestion, we assume that starting moments of consecutive vehicles at the storage area are delayed by a time units. Note that this way of operating may result in an accumulation of raw material or finished parts at the machines. In particular, raw materials will accumulate at a given machine early in the process if several deliveries are made while the first part is being manufactured. Conversely, if several short operations are performed on a machine towards the end of the production schedule, the result will be an accumulation of finished parts that will have to be collected by separate vehicle trips after the production schedule is completed. On average, however, a vehicle delivering raw material to a machine will be able to collect a finished part from the same machine, and so accumulation of raw materials and finished parts should remain low. Although our model does not explicitly include the scheduling of collection of finished parts, an allowance of a time units is made in order to allow two deliveries and two collections per vehicle trip. This assumption ensures that vehicle congestion will never occur; however, the assumption results in a slight overestimation of the time required for the vehicle schedule. This overestimation is limited by the facts that a should be small with respect to A and that the proportion of trips in which vehicles are unable to collect finished parts is also small.

The problem is now to construct a *schedule* for machines and vehicles such that the whole task set is processed in minimum time.

It is obvious that the general problem stated above is NP-hard, since it is already NP-hard for the nonpreemptive scheduling of two machines (cf. Coffman 1976). In the following two sections, we will consider two variants of the problem. In the first, the production schedule (i.e., the assignment of operations to machines) is assumed to be known, and the objective is to find a feasible schedule for vehicles. This problem can be solved in polynomial time. The second problem consists of finding a composite schedule, i.e., one taking into account the simultaneous assignment of vehicles and machines to operations.

3. Vehicle scheduling for a fixed production schedule

In this section, we consider the problem of vehicle scheduling given a production schedule. Suppose that a (possibly optimal) nonpremptive assignment of operations to machines in

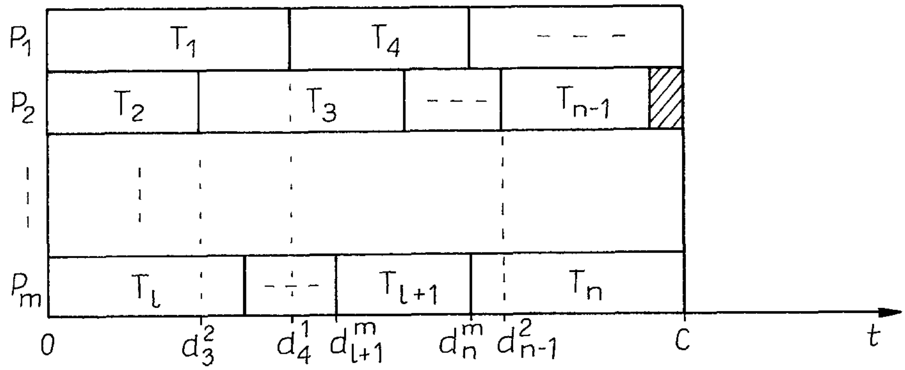

Figure 2. An example production schedule.

time is given (see figure 2). This assignment imposes certain deadlines d_j^i on the delivery of pieces of raw material to particular machines, where d_j^i denotes the latest time by which raw material for part T_j should be delivered to machine P_i. (The operation corresponding to this part starts on machine P_i at time d_j^i.) Lateness in delivery could result in exceeding the planned schedule length C. Below we describe an approach that allows us to check whether it is possible to deliver all the required parts to their destinations (given some production schedule); if so, a feasible vehicle schedule will be constructed (i.e., a feasible assignment of parts to vehicles in time). Without loss of generality, we may assume that at time 0 there is already at every machine a piece of material to produce the first part; otherwise, one should appropriately delay starting times on consecutive machines (see figure 3).

Our vehicle scheduling problem may now be formulated as follows. Given a set of deadlines d_j^i, $j = 1, 2, \ldots, n$ and delivery times from the input station to particular machines τ_i, $i = 1, 2, \ldots, m$, is it possible to deliver all the required parts on time, i.e., before the respective deadlines? If the answer is positive, a feasible vehicle schedule should be constructed. In general, this is equivalent to determining a feasible solution to a *Vehicle Routing with Time Windows* (see e.g., Desrochers et al. 1988). Let T_0 and T_{n+1} be two dummy operations representing the first departure and the last arrival of every vehicle, respectively. Also define two dummy machines P_0 and P_{m+1} on which T_0 and T_{n+1} are executed, respectively, and let $\tau_0 = 0$, $\tau_{m+1} = M$, where M is an arbitrary large number. Denote by $i(j)$ the index of the machine on which T_j is executed. For any two operations T_j, $T_{jj'}$, *let* $c_{jj'}$, be the travel time taken by a vehicle to make its delivery for operation $T_{j'}$ immediately after its delivery for T_j:

$$c_{jj'} = \begin{cases} \tau_{i(j')} - \tau_{i(j)} & \text{if } \tau_{i(j')} \geq \tau_{i(j)} \\ A - \tau_{i(j')} - \tau_{i(j)} & \text{if } \tau_{i(j')} < \tau_{i(j)} \end{cases}$$

$$(j, j' = 0, \ldots, n + 1; j \neq j').$$

If $\tau_j + c_{jj'} \leq \tau_{j'}$, define a binary variable $x_{jj'}$ equal to 1 if and only if a vehicle makes its delivery for $T_{j'}$ immediately after its delivery for T_j. Also, let u_j be a nonnegative variable denoting the latest possible delivery time of raw material for operation T_j ($j = 1, \ldots, n$). The problem then consists of determining whether there exists values of the variables satisfying

$$\sum_{j'=1}^{n} x_{oj'} = \sum_{j=1}^{n} x_{j,n+1} = k \tag{1}$$

$$\sum_{\substack{j=0 \\ j \neq l}}^{n+1} x_{jl} = \sum_{\substack{j'=0 \\ j' \neq l}}^{n+1} x_{lj'} = 1 \qquad (l = 1, \ldots, n) \tag{2}$$

$$u_j - u_{j'} + Mx_{jj'} \leq M - c_{jj'}, \qquad (j, j' = 1, \ldots, n; j \neq j') \tag{3}$$

$$0 \leq u_j \leq d_j^i \tag{4}$$

In this formulation, constraint (1) specifies that k vehicles are used, while constraint (2) associates every operation with exactly one vehicle. Constraints (3) and (4) guarantee that the vehicle schedule will satisfy time-feasibility constraints. They are imposed only if $x_{jj'}$ is defined. This feasibility problem is in general NP-complete (Savelsbergh 1985). However, our particular problem can be solved in polynomial time because we can use the cyclic property of the schedule for relatively easily checking of the feasibility condition of the vehicle schedule for a given production schedule. The first schedule does not need to be constructed. When checking this feasibility condition, one uses the operation *latest transportation starting times*, defined as follows:

$$s_j = d_j^i - \tau_i, \qquad j = 1, 2, \ldots, n.$$

The feasibility checking is given in lemma 1.

Lemma 1. For a given ordered set of latest transportation starting times s_j, $s_j \leq s_{j+1}$, $j = 1, 2, \ldots, n$, one can construct a feasible transportation schedule for k vehicles if and only if

$$s_j \geq \left(\left\lceil \frac{j}{k} \right\rceil - 1 \right) A + \left[j - \left(\left\lceil \frac{j}{k} \right\rceil - 1 \right) k - 1 \right] a$$

for all $j = 1, 2, \ldots, n$, where $\lceil j/k \rceil$ denotes the smallest integer not smaller than j/k.

Proof. It is not hard to prove the correctness of the above formula taking into account the fact that its two components reflect, respectively, the time necessary for an integer number of cycles and the delay of an appropriate vehicle in a cycle needed for a transportation of the jth operation in order. ■

The conditions given in lemma 1 can be checked in $O(n \log n)$ time in the worst case. If one wants to construct a feasible schedule, the following polynomial-time algorithm will find one whenever one exists. The basic idea behind the algorithm is to choose for transportation a part whose deadline, less corresponding delivery time, is minimum—i.e., the most urgent delivery at this moment. This approach is summarized by the following algorithm.

Algorithm 1

1. Set $t = 0$, $l = 0$.
   For all the operations (parts) to be delivered to machines, calculate their latest transportation starting times.
2. At moment t, when a vehicle is ready for loading at an input station, consider the set of remaining operations and calculate their *slack times*:
   $sl_j = s_j - t$.
   If all are nonnegative, go to step 3; otherwise stop (no feasible vehicle schedule exists).
3. Choose a part with the minimum value of sl_j and load it onto the vehicle.
   Set $l = l + 1$. If $l \leq k - 1$, then $t = t + a$; otherwise, $t = t - (k - 1)a + A$ and $l = 0$.
   If there are any operations undelivered to corresponding machines, then go to step 2; otherwise, stop. ■

A basic-property of algorithm 1 is proved in the following theorem.

Theorem 1. Algorithm 1 finds a feasible transportation schedule whenever one exists.

Proof. Suppose that algorithm 1 fails to find a feasible transportation schedule while such a schedule S exists. In this case there must exist in S two parts T_i and T_j such that $sl_i <$ sl_j and T_j has been transported first. It is not hard to see that by exchanging these two

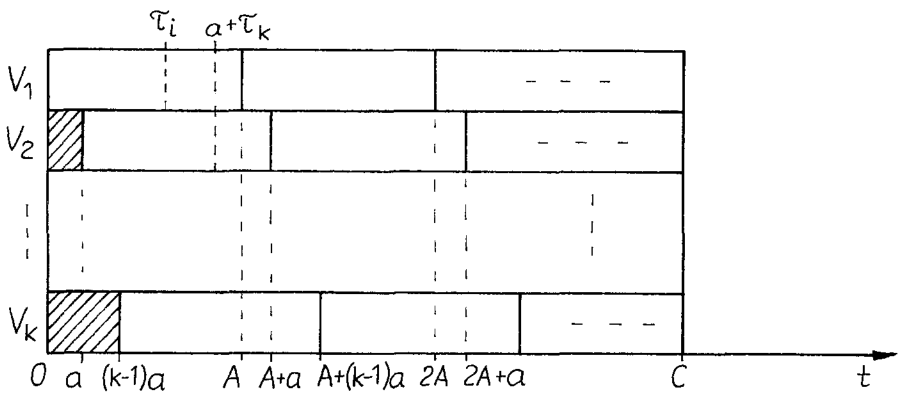

Figure 3. An example vehicle schedule.

parts, i.e., transporting T_i first, we do not cause the infeasibility of the schedule. Now we can repeat the above pattern as long as such a pair of parts violating the earliest slack-time rule exists. After a finite number of such changes, one gets a feasible schedule constructed according to algorithm 1, which is a contradiction. ■

Let us now calculate the complexity of algorithm 1 considering the off-line performance of the algorithm. Then its most complex function is the ordering of operations in nondecreasing order of their slack times. Thus, the overall complexity would be $O(n \log n)$. However, if one performs the algorithm in the on-line mode, then the selection of the operation to be transported next requires only linear time, provided that an unordered sequence is used. In both cases, a low-order polynomial-time algorithm is obtained. We see that the easiness of the problem depends mainly on its regular structure following the cyclic property of the vehicle schedule.

To illustrate the use of the algorithm, consider the following example. Let the number of machines m, the number of operations n, and the number of vehicles k be equal to 3, 9, and 2, respectively. Transportation times for respective machines are $\tau_1 = 1$, $\tau_2 = 1.5$, $\tau_3 = 2$ and cycle, and loading and unloading times are $A = 3$, $a = 0.5$, respectively. A production schedule is given in figure 4a. Thus the deadlines are $d_5^1 = 3$, $d_7^1 = 7$, $d_6^2 = 6$, $d_8^2 = 7$, $d_4^3 = 2$, and $d_9^3 = 8$. They result in the latest transportation starting times $s_4 = 0$, $s_5 = 2$, $s_6 = 4.5$, $s_7 = 6$, $s_8 = 5.5$, and $s_9 = 6$. The corresponding vehicle schedule generated by algorithm 1 is shown in figure 4b. Part T_9 is delivered too late and no feasible transportation schedule for the given production plan can be constructed.

The obvious question now is what to do if there is no feasible transportation schedule. The first approach consists of finding operations in the transportation schedule that can be delayed without lengthening the schedule. If such an operation is found, other operations that cannot be delayed are transported first. In our example (figure 4a), operation T_7 can be started later and instead T_g can be assigned first to vehicle V_1. Such an exchange will not lengthen the schedule. However, it may also be the case that the production schedule reflects deadlines that cannot be exceeded, and therefore the operations cannot be shifted. In such a situation, one may use an alternative production plan, if one exists. As pointed out in Schmidt (1989) it is often the case at the FMS planning stage that several such plans have been constructed, and the operator chooses one of them. If none can be realized because of a nonfeasible transportation schedule, the operator may decide to construct optimal production and vehicle schedules at the same time. One such approach based on dynamic programming is described in the next section.

4. Simultaneous task and vehicle scheduling

In this section, the problem of simultaneous construction of production and vehicle schedules is discussed. As mentioned above, this problem is NP-hard, although not strongly NP-hard. Thus, a pseudopolynomial-time algorithm based on dynamic programming can be constructed for its solution.

Assume that tasks are ordered in nonincreasing order of their processing times, i.e., $p_1 \geq p_2 \geq \ldots p_{n-1} \geq p_n$.

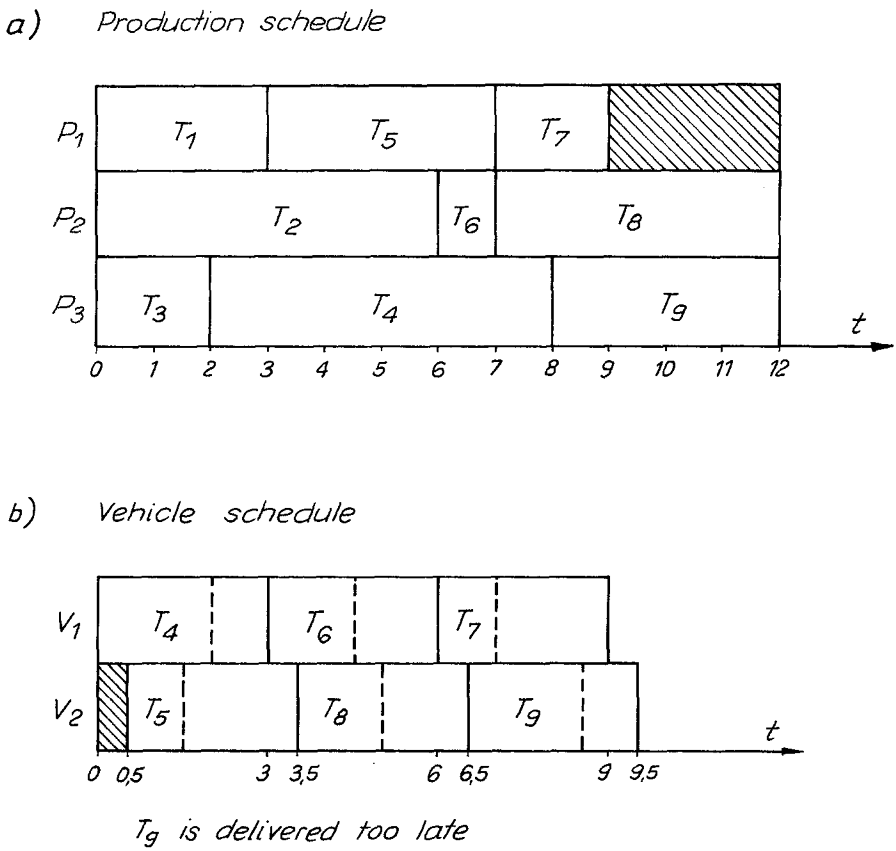

Figure 4. Production and nonfeasible vehicle schedules.

Such an ordering implies that longer tasks will be processed first and that processing can take place on machines further from the storage area—a convenient fact from the viewpoint of vehicle scheduling.

Now let us formulate a dynamic programming algorithm using the ideas presented in Graham et al. (1979).

Define

$$x_j(t_1, t_2, \ldots, t_i) = \begin{cases} \textit{true}, \text{ if operations } T_1, T_2, \ldots, T_j \text{ can be scheduled on machines } P_1, P_2, \ldots, P_m \text{ in such a way that } P_i \text{ is busy in time interval } [0, t_i],\ i = 1, 2, \ldots, m \text{ (excluding possible idle time following from vehicle scheduling) and that the vehicle schedule is feasible;} \\ \textit{false}, \text{ otherwise} \end{cases}$$

where

$$x_0\,(t_1, t_2, \ldots, t_m) = \begin{cases} \textit{true}, \text{ if } t_i = 0,\ i = 1, 2, \ldots, m \\ \textit{false}, \text{ otherwise} \end{cases}$$

Using these variables, the recursive equation can be written in the following form:

$$x_j\,(t_1, t_2, \ldots, t_m) = \bigvee_{i=1}^{m} [x_{j-1}\,(t_1, t_2, \ldots, t_{i-1}, t_i - p_j, t_{i+1}, \ldots, t_m)$$

$$\wedge\ Z_{ij}\,(t_1, t_2, \ldots, t_{i-1}, t_i, t_{i+1}, \ldots, t_m)]$$

where

$$Z_{ij}\,(t_1, t_2, \ldots, t_i, t_{i+1}, \ldots, t_m) = \begin{cases} \textit{true, if } t_i - p_j - \tau_i \geq (\lceil j/k \rceil - 1)\,A \\ \quad + [j - (\lceil j/k \rceil - 1)\,k - 1]\,a \textit{ or } j \leq m \\ \textit{false}, \text{ otherwise} \end{cases}$$

is the condition of vehicle schedule feasibility, given in lemma 1.

Values of $x_j(\cdot)$ are computed for $t_i = 0, 1, \ldots, C$, $i = 1, 2, \ldots, m$, where C is an upper bound on the minimum schedule length $C^*_{\max}$. Finally, $C^*_{\max}$ is determined as

$$C^*_{\max} = \min \{\max \{t_1, t_2, \ldots, t_m\} : x_n(t_1, t_2, \ldots, t_m) = \textit{true}\ \}$$

The above algorithm solves our problem in $O(nC^m)$ time. Thus, for fixed m, it is a pseudopolynomial-time algorithm, and can be used in practice, taking into account that m is rather small.

To complete our discussion, let us consider once more the example from section 3. The above dynamic programming approach yields schedules presented in figure 5. We see that it is possible to complete all the operations in 11 time units and deliver them to machines in 8 units.

5. Conclusions

In this article, the model of an FMS has been introduced, taking into account both machine and vehicle scheduling. Minimization of maximum operation completion time was the criterion. A simple polynomial-time algorithm can be used to find a feasible vehicle schedule (whenever one exists), provided that a production schedule is given. For a fixed number of machines, a pseudopolynomial dynamic programming approach solves both problems, constructing optimal production and vehicle schedules. Various extensions of the model are possible and worth considering. Among them are those including different routes for particular vehicles, an inspection phase as the second stage machine, resource competition, and different criteria (e.g., maximum lateness). These issues are currently under investigation.

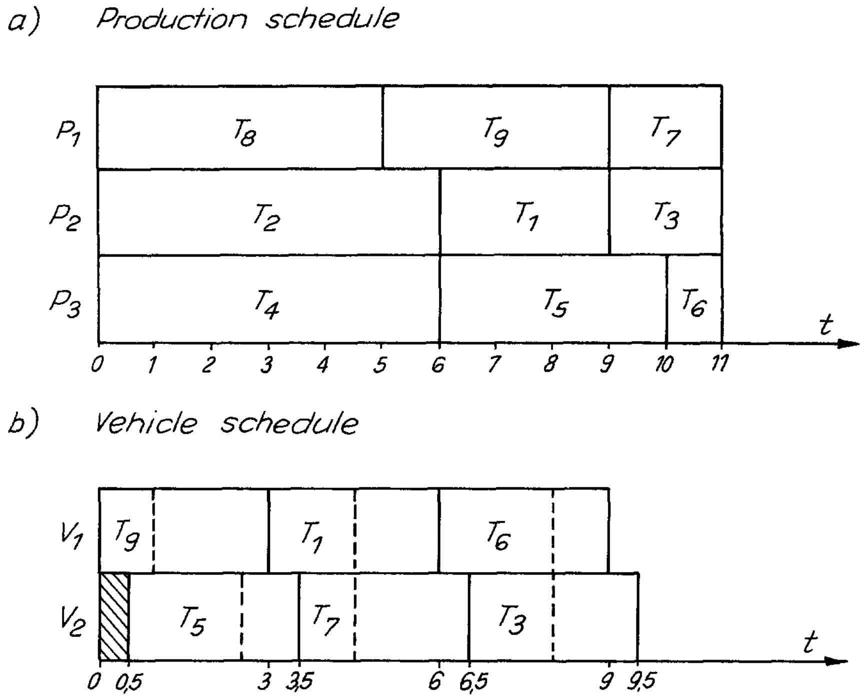

Figure 5. Optimal production and vehicle schedules.

Acknowledgment

The authors are grateful to K.E. Stecke and to anonymous referees for their comments, which greatly improved the presentation of the results contained in this article.

References

Afentakis, P., "An Optimal Scheduling Strategy for Flexible Manufacturing Systems," *International Journal Flexible Manufacturing Systems,* Vol. 1, No. 2 (1985).

Blażewicz, J., Cellary, W., Slowiński, R., and Weglarz, J., "Scheduling under Resource Constraints: Deterministic Models," *Annals of Operations Research,* Vol. 7 (1986).

Blażewicz, J., Finke, G., Haupt, R., and Schmidt, G., "New Trends in Machine Scheduling," *European Journal of Operational Research,* Vol. 37, No. 3, pp. 303–317 (1988).

Carrie, A.S. and Petsopoulos, A.C., "Operations Sequencing in an FMS," *Robotica,* Vol. 3, No. 4, pp. 259–264 (1985).

Chang, Y.L. and Sullivan, R.S., "Real-time Scheduling of FMS," paper presented at the TIMS/ORSA Meeting, San Francisco, CA (1984).

Coffman, E.G. Jr. (Ed.), *Computer and Job/Shop Scheduling Theory,* John Wiley, New York (1976).

Desrochers, M., Lenstra, J.K., Savelsbergh, M.W.P., and Soumis, F., "Vehicle Routing with Time Windows," in *Vehicle Routing: Methods and Studies,* B.L. Golden and A.A. Assad (Eds.), North-Holland, Amsterdam, pp. 65–84 (1988).

Erschler, J., Roubellat, F., and Thuriot, C., "Periodic Release Strategies for FMS," paper presented at the TIMS/ORSA Meeting, San Francisco, CA (1984).

Finke, G. and Kusiak, A., "Modeling and Solving the Flexible Forging Module Scheduling Problem," *Engineering Optimization*, Vol. 12, No. 1, pp. 1-12 (1987).

Graham, R.L., Lawler, E.L., Lenstra, J.K., and Rinnooy Kan, A.H.G., "Optimization and Approximation in Deterministic Sequencing and Scheduling Theory: A Survey," *Annals of Discrete Mathematics*, Vol. 5, pp. 287-326 (1979).

Jaikumar, R., "Postindustrial Manufacturing," *Harvard Business Review*, Vol. 64, No. 6, pp. 69-76 (December 1986).

Jaikumar, R. and Van Wassenhove, L.N., "A Production Planning Framework for Flexible Manufacturing Systems," working paper, Harvard Business School (1987).

Kusiak, A., "Application of Operational Research Models and Techniques in Flexible Manufacturing Systems," *European Journal of Operational Research*, Vol. 24, No. 3, pp. 336-345 (1986a).

Kusiak, A (Ed.), *Flexible Manufacturing Systems: Methods and Studies*, North-Holland, Amsterdam (1986b).

Kusiak, A. (Ed.), *Modelling and Design of Flexible Manufacturing Systems*, Elsevier, New York (1986c).

Kusiak, A., "Scheduling Flexible Machining and Assembly Systems," *Annals of Operations Research*, Vol. 15, pp. 337-352 (1988).

Kusiak, A. and Wilhelm, W.E. (Eds.), "Analysis, Modelling and Design of Modern Production System," *Annals of Operations Research*, Vol. 17 (1989).

Savelsbergh, M.W.P., "Local Search for Routing Problems with Time Windows," *Annals of Operations Research*, Vol. 4, pp. 285-305 (1985).

Schmidt, G., *CAM: Algorithmen und Decision Support für die Fertigungssteuerung*, Springer Verlag, Berlin (1989).

Srishkandarajah, C., Sethi, S.P., and Ladet, P., "Scheduling Methods for a Class of Flexible Manufacturing Systems," *Annals of Operations Research*, Vol. 17, pp. 139-162 (1989).

Stecke, K.E., "Design, Planning, Scheduling and Control Problems of Flexible Manufacturing Systems," *Annals of Operations Research*, Vol. 3, pp. 3-12 (1985).

Stecke, K.E. and Solberg, J.J., "Loading and Control Policies for a Flexible Manufacturing System," *International Journal of Production Research* Vol. 19, No. 5, pp. 481-490 (September-October, 1981).

Stecke, K.E. and Suri, R. (Eds.), "Flexible Manufacturing Systems: Operations Research Models and Applications," *Annals of Operations Research*, (1985).

Stecke, K.E. and Suri, R. (Eds.), "Flexible Manufacturing Systems: Operations Research Models and Applications II," *Annals of Operations Research*, (1988).

Villa, A. and Rosetto, S., "On the Joint Problem of Dynamic Part Routing and Station Service Control in Flexible Manufacturing Systems," *International Journal of Material Flow*, Vol. 2, pp. 97-110 (1985).

Yao, D.D., "Material and Information Flow in Flexible Manufacturing Systems," *International Journal of Material Flow*, Vol. 2, pp. 143-149 (1985).