

SCHEDULING PRODUCTION ORDERS, TAKING INTO ACCOUNT DELAYS AND WASTE

Robert Dylewski¹, Andrzej Jardzioch²

¹ *University of Zielona Góra, Faculty of Mathematics, Computer Science and Econometrics, Poland*

² *West Pomeranian University of Technology Szczecin, Faculty of Mechanical Engineering and Mechatronic, Poland*

Corresponding author:

Andrzej Jardzioch

West Pomeranian University of Technology Szczecin

Faculty of Mechanical Engineering and Mechatronic

Al. Piastów 19, 70-310 Szczecin, Poland

phone: +48 91 449 44 78

e-mail: Andrzej.jardzioch@zut.edu.pl

Received: 5 May 2014

Accepted: 8 July 2014

ABSTRACT

The article addresses the problem of determining the sequence of entering orders for production in a flexible manufacturing system implementing technological operations of cutting sheet metal. Adopting a specific ranking of production orders gives rise to the vector of delays and waste in the form of incompletely used sheets. A new method was postulated for determining the optimal sequence of orders in terms of two criteria: the total cost of delays and the amount of production waste. The examples illustrate the advantages of the proposed method compared with the popular heuristic principles.

KEYWORDS

flexible manufacturing systems, production control, scheduling algorithms.

Introduction

According to the current paradigm of agile manufacturing, the most important task for manufacturing organizations is to meet customer expectations by ensuring timely execution while maximizing profit [1–4]. This task can be accomplished by fully integrating management, planning and control subsystems. A practical solution to such requirements is a flexible manufacturing system, allowing for the manufacture of products in an automatic cycle while maintaining the possibility of rapid adjustment to changing customer needs [3, 4]. While analyzing the operation of flexible production systems, it should be noted that the effective work decisively depends on the proper operation of the control subsystem and more specifically on the control algorithms implemented in the subsystem. It is thanks to these algorithms, client orders are scheduled in the correct order, items from warehouses are taken, transport trucks are controlled, and control and diagnostics operations are performed [6, 7]. One of the most im-

portant tasks solved by the algorithms controlling the operation of flexible manufacturing systems is to determine the order of tasks provided for production (Scheduling problem) [8, 9]. An analysis of the operation of the post-production systems leads to the conclusion that the order of placing tasks affects the broadly defined efficiency of the whole production system. This efficiency is understood as ensuring both customer satisfaction by minimizing delay times of the orders executed, and taking into account economic and environmental aspects of the production process. One of the most important parameters affecting the efficiency and environmental aspects of the production process is the amount of waste [10].

The problems associated with the development of efficient algorithms to control the operation of flexible manufacturing systems are subject to research by many scientific papers. Two concepts are often mentioned: dynamic scheduling and reactive scheduling [11].

Dynamic scheduling is based on the successive selection of orders entered into production based on

the current information about ongoing production processes [11, 12]. This information may include, e.g. the time of completion of processing for each individual machine, the stock in station-side storages, tool changing times, equipment failures, delays in delivery of material and the results of interoperative control. Due to the very strict time limits, heuristic method and simple priority rules are most often used in this case [11].

The idea of reactive scheduling is to prepare the entire production schedule for stationary conditions and then modify it in the event of unforeseen events [11, 13].

Particular interest is invested in developments taking into account the need to obtain acceptable solutions in an actual industrial environment. In [3] a methodology for real-time design of production schedules with the use of priority rules and simulation studies has been presented. One may also notice a tendency to include more than one criterion, which allows obtaining solutions more suited to conditions prevailing in production systems [7, 14, 15].

In practice, production planning often involves priority rules due to their simple structure and low computational complexity [4, 12]. One can mention such rules as: a rule scheduling orders according to their date of entry in the system (FIFO), a rule scheduling orders by the date of their execution (EDD), a rule scheduling orders according to the shortest processing time (SPT), a rule scheduling orders according to the total time of operations left to perform (LWR), a rule scheduling orders by their assigned delay penalties (DDP).

In terms of the NP-hard nature of the problems associated with scheduling production orders often, the use of solutions applying tabu search approach, genetic, immune algorithms and neural networks is proposed [5, 16, 17]. Due to the time limits, the methods applicable in this case do not include such exact methods as branch-and-bound (B&B) or integer linear programming (ILP). On the other hand, algorithms based on evolutionary methods do not guarantee reaching optimal solutions and therefore may prevent the full utilization of a system's production capacity [18]. Therefore, the application taboo search algorithms, which allow receiving the optimal solutions in acceptable time, may be particularly promising.

The article presents an algorithm which allows finding ε -optimal solutions for scheduling production orders in a flexible manufacturing system implementing technological operations of cutting sheet metal. An ε -optimal solution in this case stands for a solution that is not worse than the optimal one by more

than a predetermined value ε (for $\varepsilon = 0$, an ε -optimal solution is optimal). Adopting a specific ranking of production orders causes the appearance of the delay vector and waste in the form of incompletely used sheets. A new method to determine the optimal sequence of orders has been proposed in terms of two criteria: the total cost of delays and the amount of production waste. The examples illustrate the advantages of the proposed method compared with the simple heuristic rules.

Description of the problem of scheduling production orders

The technological process carried out in an automated, flexible manufacturing system, which implements cutting sheet metal components has been shown in Fig. 1. The most important element of the analyzed system is the M1 jet cutting machine and the automated M_{in} racking sheet store, which stores various sheets provided as input material to the production process. Production orders are passed by the client to the parent planning subsystem via the Internet, and are then automatically scheduled and subsequently transferred for implementation. After processing, the executed orders are stored in the M_{out} output store. In terms of technical limitations associated with the total automation of the process of generating the code for CNC jet cutting, it has been assumed that the input store will only hold full sheets of metal with standard dimensions 1200×800 mm. Partly used sheets are treated as waste, and their possible further use takes place outside the analyzed process.

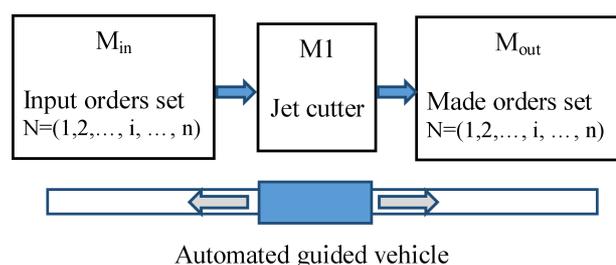


Fig. 1. Flexible manufacturing system.

Based on preliminary simulation research, it can be concluded that the sequence in which orders are transmitted to the production system largely affects not only the completion date of the various production orders but also the volume of waste sheets used in the process.

The main task of the process of scheduling production orders is determining their sequence of execution in such a way as to prevent delays in their im-

plementation. Unfortunately, this is not always possible. However, delays are common in the production practice in the execution of manufacture orders, usually associated with the desire to receive too many orders and setting unrealistic deadlines for their implementation. Failure to meet the deadlines for production orders can be costly. Most contracts for the execution of an order also include provisions on penalties related to untimely performance. These penalties are sometimes severe and may adversely affect the financial results of the company. Therefore, while setting the sequence of production orders, attention should be paid first and foremost to minimizing the costs associated with delayed execution of orders. This issue has been presented in detail in [19]. This paper also proposes a method, which allows minimizing delays for cutting sheet metal in a flexible manufacturing system.

Another equally important issue in the planning process for automatic cutting of sheet metal is the problem of waste, defined as metal residue, which can no longer be used for cutting the next job. Two cases may be defined. The first occurs when, as a result of the adopted schedule, the next job requires sheets of different thickness than the one required for previous orders. In this case, regardless of the size of the area still available for use, it is necessary to replace the sheet. The second case occurs when the dimensions of objects in the next job are so large (or so unusual) that they cannot be cut on the surface remaining after the execution of the preceding order. In such a situation, the partially cut sheet is also removed from the system and a new sheet must enter from the store in its place. To better illustrate this problem, we can describe an example where the optimal schedule adopted based on the criterion of minimizing delays will require replacing the sheet after each job. In this case, even though orders will be executed with minimal delays, there will be large losses in the system associated with the use of a small area of the stored sheets. In addition to the associated direct material costs, such action could lead to further delays in the execution of orders in terms of the lack of new sheets in the store. In order to minimize the waste, a procedure can be applied to connect the orders within groups, so as to allow cutting from the same sheet. This method can, however, lead to high costs related to breaching the scheduled production deadlines.

Noticing the ability to control the size of the waste depending on the accepted sequence of production orders resulted in the need to develop procedures for scheduling orders, taking into account two criteria: the total delay times and total waste.

Scheduling production orders, taking into account the cost of delays and total waste

This section proposes a method to determine the sequence of production orders, taking into account delay times of orders and total waste. It consists of four main stages. In a first stage, the base sequence is determined, which gives the smallest maximum delays per job. In the second stage, the sequence of orders giving the minimum total delay is determined. In the third stage, for a fixed ε , the ε -optimal sequences are determined, i.e. those with the sum of delays no greater than the solution with the minimum sum of the delays plus ε . In the fourth step, ε -optimal solutions are searched through for a schedule capable of producing as little waste as possible.

It is assumed that for a given list of orders (z_1, z_2, \dots, z_n) , the following information is available:

$t_o(i)$ – processing time of the order z_i ,

$t_t(i)$ – deadline for order z_i ,

$ob(i)$ – area required to cut the job.

Let S_o mean the total processing time of a given list of n orders:

$$S_o = \sum_{i=1}^n t_o(i). \quad (1)$$

One of the simplest methods is the EDD method, where the orders are sorted by increasing deadline for completion t_t (from the one with the earliest deadline to the one with the latest deadline).

Maximum delays for a single order in the sequence obtained by this method are not greater than in all the other sequences. Unfortunately, this sequence does not necessarily give the smallest sum of delays (see example Table 2).

This section proposes a method that ensures appointing ε -optimal sequences of orders in terms of the minimum amount of delay, i.e. the optimal sequence giving the minimum amount of delays and sequences worse than optimal by no more than the set ε value. This method is a generalization of the MOpt method proposed by the authors in [19], i.e. for $\varepsilon = 0$ the presented method is used to determine optimal solutions, such as in MOpt. Determination has been divided into three stages. In the first step, a sequence is obtained according to the one obtained using the EDD method obtained. In the second step, the optimal sequences are obtained in terms of the minimum total delay, and in the third - ε -optimal sequences. Let us denote the presented method an MOpt- ε .

MOpt- ε method:

Determination of the solutions is in the form of a tree.

Stage I:

At the beginning, in block 1 (the root of the tree), for each order a number is determined

$$p(i) = S_o - t_i(i), \quad (2)$$

which determines the delay in the case, when the order z_i is done as a last. If $p(i) \leq 0$ for every $i = 1, \dots, n$, this means that every one of all possible sequences gives total delay equal 0 (end). In the case where not every $p(i)$ is less than or equal 0, the order z_j is first selected, where $p(j)$ is the smallest, and placed at the end of the queue. This order has a predetermined time required to process the remaining orders

$$S'_o := S_o - t_o(j) \quad (3)$$

and the delay that arises as a result of placing the order at the end

$$op = \max \{0, p(j)\}. \quad (4)$$

Then, in block 2 (the successor of block 1 in the solution tree), the same is performed as in the root of the tree, although skipping the order that has already been put at the end of the queue. The currently selected order z_k is put into the queue as the second last, updating the time S'_o ($S'_o := S'_o - t_o(k)$) and the total delay op ($op := op + \max \{0, p(k)\}$).

Then, the operations performed in block 2 are repeated until reaching block n (leaf in the tree), from which the order is put at the beginning of the queue. This way the base branch in the tree is obtained (with preset sequence of all orders), for which the sequence of orders is in line with the one acquired using the EDD method. If the sum of delays for this branch $Sop = 0$, then the resulting sequence is optimal in terms of total delay.

Stage II:

If the base branch $Sop > 0$, then since block 1 it is verified whether selecting the next order (in terms of the minimum $p(i)$), would result in exceeding the sum of delays Sop from the base solution. If so, another branch will not be formed. If not, the selected order is inserted into the appropriate place in the queue in the next block (the successor of block 1), etc.

If another leaf in the tree is reached (with a preset sequence of all orders), the sum of delays in this sequence is not greater than the one in the base solution. The Sop is therefore updated and further branches are verified until no more can be developed (Sop is exceeded). Finally, the optimal solutions are

found in the leaves with the lowest Sop possible; let us mark them as Sop_m .

Stage III:

If $\varepsilon > 0$, it is verified in blocks, which are not leaves, whether selecting the next order (in terms of the minimum $p(i)$ value) will not result in exceeding $Sop_m + \varepsilon$. If so, this branch is not developed any further. If not, the selected task is set in the next block in the appropriate place relative to the end of the queue, etc.

If we reach new leaves of the tree, the total delay corresponding to the sequences of these leaves are not greater than $Sop_m + \varepsilon$. Finally, ε -optimal solutions are found in the leaves, where $Sop \leq Sop_m + \varepsilon$.

Using the algorithm proposed below, one can determine the total waste and the number of used sheets for ε -optimal sequences determined using the MOpt- ε method.

Stage IV:

The algorithm for determining total waste and the number of sheets used.

Designations:

ob – area used for the cut sheet,

reserve – free area in the cut sheet, (ob + reserve = 100%)

waste – total waste from cut sheets,

sheets – number of sheets used,

n – number of orders,

$obi = ob(U(i))$ – area required to cut the i -th order in the given sequence (in %), $i = 1, \dots, n$,

Algorithm:

ob := 0; reserve = 100%; waste = 0; sheets := 1;

for i:=1 to n do

obi := ob(U(i));

if obi \leq reserve then

ob := ob + obi;

reserve := reserve – obi;

else

waste := waste + reserve;

sheets := sheets + 1;

ob := obi;

reserve := 100% - obi;

waste := waste + reserve;

Example

Data for the example is presented in Table 1 This example assumes four orders, for which the processing time t_o has been defined, as well as the scheduled deadline t_t , slack time of performance z ($z = t_o - t_t$) and the area of cut sheet o_b required to carry out the order relative to the overall surface of the sheet. It was assumed that individual orders consist of single

items, which means that each order needs to be cut from a single metal sheet.

Table 1
Data for the example.

Orders	Processing time t_o	Deadline t_t	Slack time z	Sheet area o_b
	[min]	[min]	[min]	[%]
z_1	10	150	140	50
z_2	20	30	10	40
z_3	100	110	10	20
z_4	50	60	10	70

To determine ε -optimal sequences, the MOpt- ε method was used, as presented in p. 3. In the first stage, the base sequence U_1 (z_2 z_4 z_3 z_1) was established, where the total delay was set at $Sop = 100$ min. This is a sequence ordered by increasing deadlines. Then, in the second stage, the optimal sequence has been determined in terms of total delay U_2 (z_2 z_4 z_1 z_3) with the sum of delays $Sop_m = 80$ min. In the third stage a tolerance of $\varepsilon = 10$ min has been assumed, where ε means the limit value, by which the ε -optimal solution may differ from the optimal one in terms of the total cost of delays. In this step, two more sequences were determined: U_3 and U_4 , where $Sop \leq Sop_m + \varepsilon$ (see Table 2). Finally, out of the four sequences established during the initial three stages, sequences U_2 , U_3 and U_4 are ε -optimal solutions for a given ε .

These three schedules were classified into the fourth stage, where total waste and the number of sheets used (last two columns in Table 2) was determined using the presented algorithm (Stage IV). Table 2 also gives the amount of waste and the number of sheets used in sequence U_1 (it is not ε -optimal for $\varepsilon = 10$ min). As it turns out, sequences U_3 and U_4 result in waste of about 20% and 2 used sheets. To implement a sequence with as little delay as possible U_2 , 3 sheets are required, and the waste will be greater by a whole sheet compared to sequences U_3 and U_4 . In the optimal solution U_2 , the cutting process begins with the order z_2 . Since the next order in the sequence is z_4 , which required 70% of the sheet's area, it was necessary to enter a new sheet, which resulted in 60% waste from the first sheet. After cutting the order z_4 , it was necessary to enter a new sheet, which resulted in 30% waste. The next two orders z_1 and z_3 could be cut from a single sheet with 30% waste. In total, sequence U_2 therefore requires using three sheets and generates total waste of 120% of the total sheet area.

The improper sequence of orders may give rise to a lot more total delays than the optimal solution –

even as much as 150 min, e.g. for U_5 (z_2 z_3 z_4 z_1) as well as much more waste than the best of ε -optimal solutions in terms of this criterion (for U_5 , total waste is 1.2 sheets, as in the case of U_1 and U_2).

Table 2
Determined sequences.

Sequence of orders	Total delay	Total waste	Number of sheets
U_1 (z_2, z_4, z_3, z_1)	0 + 10 + 60 + 30 = 100	60% + 0% + 10% + 50% = 120%	3
U_2 (z_2, z_4, z_1, z_3)	0 + 10 + 0 + 70 = 80	60% + 30% + 0% + 30% = 120%	3
U_3 (z_1, z_2, z_4, z_3)	0 + 0 + 20 + 70 = 90	0% + 10% + 0% + 10% = 20%	2
U_4 (z_2, z_1, z_4, z_3)	0 + 0 + 20 + 70 = 90	0% + 10% + 0% + 10% = 20%	2

Summary

Choosing the order of the tasks executed significantly affects the efficiency of a flexible production system. Simple methods for determining the order of tasks, dependent only on the processing time (e.g. SPT method) or the deadline of the order (e.g. method EDD) do not guarantee an optimal solution due to the sum of delays. These methods also do not allow for assessment as to how the solution reached is worse than the optimum. At the cost of slightly more difficult implementation, while taking into account the processing times and deadlines, the method proposed in the article does not have these major flaws.

First of all the method proposed in the article allows determining the optimal sequence of orders in terms of total delays. The resulting waste is also an important factor in determining the sequence. The method thus also allows determining sequences, which are not optimal and result in total delays greater than the optimal solution, but remaining under the fixed time limit at most (ε -optimal solutions).

As shown in the presented examples, by taking into account the ε -optimal solutions in terms of the sum of delays, one does not lose solutions that are

much better in terms of the other criterion - the amount of waste generated.

Taking into account the ε -optimal solutions is also justified by the fact that the cost of delays for an ε -optimal solution is often equal or slightly greater than those for the optimal solution. However, the cost related to the amount of resulting waste is important. Therefore, the proposed method allows obtaining a compromise between the cost of delays and the cost of waste.

In a further study, the authors plan to develop a method that would take into account the costs of delays instead of the delay times.

References

- [1] Elmoselhy Salah A.M., *Hybrid lean-agile manufacturing system technical facet, in automotive sector*, Journal of Manufacturing Systems, 32, 4, 598–619, 2013.
- [2] Inmana Anthony R., Saleb Samuel R., Green Kenneth W. Whitten Dwayne Jr., *Agile manufacturing: Relation to JIT, operational performance and firm performance*, Journal of Operations Management, 29, 4, 343–355, 2011.
- [3] Jeong K.-C., Kim Y.-D., *A real-time scheduling mechanism for a flexible manufacturing system: Using simulation and dispatching rules*, International Journal of Production Research, 36, 9, 2609–2626, 1998.
- [4] Ahram Tareq Z., Karwowski W., *Engineering sustainable complex systems*, Management and Production Engineering Review, 4, 4, 4–14, 2013.
- [5] Byung Jun Joo, Yong Chan Choi, Paul Xirouchakis, *Dispatching Rule-based Algorithms for a Dynamic Flexible Flow Shop Scheduling Problem with Time-dependent Process Defect Rate and Quality Feedback*, Procedia CIRP, 7, 163–168, 2013.
- [6] Krishnan M., Chinnusamy T.R., Karthikeyan T. *Performance Study of Flexible Manufacturing System Scheduling Using Dispatching Rules in Dynamic Environment*, Procedia Engineering, 38, 2793–2798, 2012.
- [7] Amol S., Metha N.K., Jain P.K., *Multicriteria dynamic scheduling by swapping of dispatching rules*, International Journal of Advanced Manufacturing Technology, 34, 9–10, 988–1007, 2007.
- [8] Prakash A., Chan F.T.S., Deshmukh S.G., *FMS scheduling with knowledge based genetic algorithm approach*, Expert Systems with Applications, 38, 3161–3171, 2011.
- [9] Lee K., Leung J. Y.-T., Jia Z.-H., Li W., Pinedo M. L., Lin B.M.T., *Fast approximation algorithms for bi-criteria scheduling with machine assignment costs*, European Journal of Operational Research, 238, 54–64, 2014.
- [10] Lin Y.-K., Chang P.-C., Chen J. C., *Reliability evaluation for a waste-reduction parallel-line manufacturing system*, Journal of Cleaner Production, 35, 93–101, 2012.
- [11] Sinreich D., Shnits B. *A Robust FMS Control Architecture with an Embedded Adaptive Scheduling Mechanism*, Journal of Manufacturing Systems, 25, 4, 301–312, 2006.
- [12] Ouelhadj D., Petrovic S., *A survey of dynamic scheduling in manufacturing systems*, Journal of Scheduling, 12, 4, 417–431, 2009.
- [13] Pacha C., Bergera T., Salleza Y., Bontea T., Adama E., Trentesauxa D., *Reactive and energy-aware scheduling of flexible manufacturing systems using potential fields*, Computers in Industry, 65, 3, 434–448, 2014.
- [14] Hoogeveen H., *Multicriteria scheduling*, European Journal of Operational Research, 167, 3, 592–623, 2005.
- [15] Solimanpura M., Elmi A., *A tabu search approach for cell scheduling problem with makespan criterion*, International Journal of Production Economics, 141, 2, 639–645, 2013.
- [16] Jardzioch A., Skobiej B., *Analysis of Variable Changeover Times Impact on the Revenue in Manufacturing Process*, Academic Journal of Manufacturing Engineering, 11, 114–117, 2013.
- [17] Józefowska J., Zimniak A., *Optimization tool for short-term production planning and scheduling*, International Journal of Production Economics, 112, 1, 109–120, 2008.
- [18] Smutnicki C., *Biomimetic Optimizers for Job Scheduling*, Computer Aided System Theory – EUROCAST 2011, Springer-Verlag, Berlin Heidelberg, pp. 195–202, 2012.
- [19] Dylewski R., Jardzioch A., *Optimization of the sequence of production, orders on account of minimum delay* [in Polish: *Optymalizacja kolejności zleceń produkcyjnych ze względu na minimalną sumę opóźnień*], Pomiary Automatyka Kontrola, 6, 527–529, 2012.