ELSEVIER

European Journal of Operational Research 115 (1999) 173–178

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

Theory and Methodology

Single-machine scheduling with learning considerations

Dirk Biskup *

Faculty of Economics and Business Administration, University of Bielefeld, Postfach 10 01 31, 33501 Bielefeld, Germany

Received 1 February 1998; accepted 1 June 1998

Abstract

The focus of this work is to analyze learning in single-machine scheduling problems. It is surprising that the well-known learning effect has never been considered in connection with scheduling problems. It is shown in this paper that even with the introduction of learning to job processing times two important types of single-machine problems remain polynomially solvable.

Keywords: Scheduling; Learning

1. Introduction

Scheduling problems have received considerable attention for many years. For most of them, the processing times are assumed as given (and sequence independent), see, for example, assumptions C2 and C3 of Baker (1974). During the last couple of years these assumptions have been relaxed in different directions; to incorporate more aspects important in practical production planning problems have been studied in which the release times of the jobs are related to the amount of resources consumed, see, for example, Li et al. (1995) and Vasilev and Foote (1997). Another step to bridge the gap between scheduling theory and practice is done by considering compressible processing times, see Vickson (1980a, b), Panwalkar and Rajagopalan (1992), Alidaee and Ahmadian (1993), Cheng et al. (1996a, b) and Biskup and Cheng (1997a). This way the normal processing time of a job can be reduced up to a minimum value by incurring higher processing costs.

A different approach to decreasing processing times is due to the concept of learning. A steady decline in processing times usually takes place by performing the same task repeatedly. Among pioneers to discover this learning effect was Wright (1936). It has received considerable attention in management science. For example, in short-term production planning its role in lotsizing is analyzed by many researchers, see, for a good literature review, Li and Cheng (1994).

The learning effect in scheduling may arise in a company which produces similar jobs on one machine or on parallel and identical machines for a number of customers. In many cases jobs will

* Tel.: 49 521 106 3929; fax: 49 521 106 6036; e-mail: dbiskup@wiwi.uni-bielefeld.de.

PII: S 0 3 7 7 - 2 2 1 7 (9 8) 0 0 2 4 6 - X

have different normal processing times due to varying (order) quantities or slightly different components that make up the products. Nevertheless, by processing one job after the other the skills of the workers continuously improve, e.g. the ability to perform setups faster, to deal with the operations of the machines and software or to handle raw materials, components or similar operations of the jobs at a greater pace.

These learning effects have been observed in numerous practical situations in different branches of industry and for a variety of corporate activities, see, for example, the literature review in Yelle (1979). It was Nadler and Smith (1963) especially who analyzed the learning curves in machine shops and obtained the result, that each basic operation used in the manufacture of a product has its own learning function. Taking further into account that due to the technology progress, shortening life cycles and an increasing diversity of products the manufacturing environment is changing incessantly, see, for example, Higgins et al. (1996), it becomes obvious that learning is not a static thing which takes place on the first few jobs to be processed. Rather than that learning needs to be considered as a constantly occurring and dynamic feature. The effect of learning has – to the best of my knowledge – never been investigated in the context of scheduling problems.

Meeting due dates and gaining short flowtimes are among the most important goals in scheduling practice, see Panwalkar et al. (1973) or Schmenner (1988). Hence, after introducing some general assumptions in Section 2, I will concentrate on these two objectives.

2. Assumptions

There are n jobs available at time zero. Each job has a normal processing time and the jobs are indexed according to the shortest (normal) processing time (SPT) rule, i.e. $p_1 \leqslant p_2 \leqslant \cdots \leqslant p_n$. The normal processing time of a job is incurred if the job is scheduled first in a sequence. The processing times of the following jobs are smaller than their normal processing times because of the learning effect. The learning curve is assumed to exist in its most popular form, which means that the time needed to perform an operation decreases by the number of repetitions, see, for example, Nadler and Smith (1963) or Yelle (1979). Let p_{ir} be the processing time of job i if it is scheduled in position r in a sequence. Then

$$p_{ir} = p_i r^a, \tag{1}$$

where $a \leqslant 0$ is the learning index, given as the logarithm to the base 2 of the learning rate. It is easy to calculate the processing times for the different jobs according to their positions in a sequence, see Table 1.

Table 1
Matrix of processing times

	$r=1$	$r=2$	...	$r=n$
p_1	p_1	$p_1 2^a$		$p_1 n^a$
p_2	p_2	$p_2 2^a$		$p_2 n^a$
...				
p_n	p_n	$p_n 2^a$		$p_n n^a$

Note that the learning effect of job i in position r only depends on its processing time, the learning index and the number of jobs processed prior to job i. The processing times of the jobs occupying the positions $1, 2, \ldots, r-1$ do not have any influence on the learning curve. Hence, the above formulation of the learning effect (1) implies that learning primarily takes place as a result of repeating 'processing time independent' operations like setups, controlling and operating machines, reading data, etc. Nevertheless, this seems to be a realistic assumption for many real life applications, especially for high-technology manufacturing processes.

3. Minimal deviation from a common due date

This section is divided into two parts. Firstly the unrestricted common due date problem with earliness, tardiness and completion time penalties is introduced and a polynomially bounded solving procedure is given. For an introduction, see Baker and Scudder (1990) or Biskup and Cheng (1997b). Proofs for the properties mentioned further down are given by Panwalkar et al. (1982), see also

Panwalkar and Rajagopalan (1992). Secondly the problem is extended by the introduction of learning effects and it is shown that it remains polynomially solvable as an assignment problem.

The goal of the unrestricted common due date problem is to jointly minimize the weighted earliness, tardiness and completion time penalty. An unrestricted common due date d is a decision variable whose value is to be determined. Note that the unrestricted common due date has no influence on the sequence of the jobs. Let $C_i, E_i = \max\{0, d - C_i\}$ and $T_i = \max\{0, C_i - d\}$ be the completion time, earliness and tardiness of job $i, i = 1, \ldots, n$, respectively. Further, let α, β and θ be the per time unit penalties for earliness, tardiness and the completion time. Then the general objective is to find a schedule S which minimizes

$$f(S) = \sum_{i=1}^{n} (\alpha E_i + \beta T_i + \theta C_i). \tag{2}$$

For example, with $\alpha = \beta = 1$ and $\theta = 0$, the objective function (2) becomes that for the well-known problem of Kanet (1981). Independent of the numerical values of α, β and θ,

$$\gamma_r = \min\{(r-1)\alpha + (n-r+1)\theta, \\ (n-r+1)(\beta+\theta)\}$$

is the positional weight which arises if a job occupies the rth position in a sequence, $r = 1, \ldots, n$. Note that these positional weights only hold if the due date is unrestricted. Consequently, Eq. (2) can be rewritten as

$$f(S) = \sum_{r=1}^{n} \gamma_r p_{[r]}, \tag{3}$$

where $[r]$ indicates the job scheduled in the rth position of a sequence. Furthermore, an optimal schedule without idle times exists, in which the bth job is completed at the due date where b is the smallest integer greater than or equal to $n\beta/(\alpha + \beta)$.

The problem (2) can be solved in two steps: Firstly a (polynomially bounded) matching algorithm to obtain an optimal sequence of the jobs by minimizing (3) is applied: The longest job is assigned to the position with the smallest weight, the second longest job to the position with the second smallest weight, etc. Secondly, the common due date is determined by summing up the processing times of the first b jobs of the optimal sequence. Thus $d = \sum_{i=1}^{b} p_{[i]}$ and the processing of the first job is started at time zero.

By introducing the learning effects discussed above, the unrestricted common due date problem can be solved as an assignment problem. Let x_{ir} be the decision variable which assumes the value 1 if job i is the rth job processed and $x_{ir} = 0$ otherwise. The objective is to find a sequence which minimizes (3), that is:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n}\sum_{r=1}^{n} \gamma_r p_{ir} x_{ir} \\ \text{subject to} \quad & \sum_{i=1}^{n} x_{ir} = 1, \quad r = 1, \ldots, n, \\ & \sum_{r=1}^{n} x_{ir} = 1, \quad i = 1, \ldots, n, \\ & x_{ir} \in \{0, 1\}, \quad i, r = 1, \ldots, n. \end{aligned}$$

The assignment problem only leads to an optimal sequence of the jobs, thus it can be compared with the first step of the algorithm mentioned above. The optimal due date has to be determined by the second step of the algorithm. Note that this second step remains unchanged since the value of b is independent of the processing times of the jobs and hence the introduction of learning effects does not have an influence on the number of early and tardy jobs. As solving an assignment problem takes $O(n^3)$ time, see, for example, Papadimitriou and Steiglitz (1982), the unrestricted common due date problem with learning effects is polynomially solvable.

The application of the algorithm is demonstrated in Example 1. For the sake of simplicity, this is an example with only four jobs. Furthermore, all numbers are rounded to one decimal, thus small rounding errors might occur.

Example 1. $n = 4$, $p_1 = 7$, $p_2 = 3$, $p_3 = 5$ and $p_4 = 4$, $\alpha = 3$, $\beta = 4$ and $\theta = 2$, the common due date d is a decision variable who's optimal value is to be determined. Learning takes place by the 80%-learning curve, thus $a = -0.322$.

Table 2
The matrix of the processing times of Example 1

	$r=1$	$r=2$	$r=3$	$r=4$
Job 1	7	$7 \cdot 2^{-0.322}=5.6$	$7 \cdot 3^{-0.322}=4.9$	$7 \cdot 4^{-0.322}=4.5$
Job 2	3	$3 \cdot 2^{-0.322}=2.4$	$3 \cdot 3^{-0.322}=2.1$	$3 \cdot 4^{-0.322}=1.9$
Job 3	5	$5 \cdot 2^{-0.322}=4$	$5 \cdot 3^{-0.322}=3.5$	$5 \cdot 4^{-0.322}=3.2$
Job 4	4	$4 \cdot 2^{-0.322}=3.2$	$4 \cdot 3^{-0.322}=2.8$	$4 \cdot 4^{-0.322}=2.6$

The processing times according to their positions are given in Table 2.

The positional penalties are $\gamma_1=8$, $\gamma_2=9$, $\gamma_3=10$, $\gamma_4=6$ with $b=3$. The optimal solution of the assignment problem is $x_{14}=x_{21}=x_{33}=x_{42}=1$, with an objective function value of 114.8. Starting the processing of the first job at time zero, the optimal due date is determined by $d=3+3.2+3.5=9.7$. The optimal schedule is shown in Fig. 1.

4. Minimal flowtime

One of the elementary results of single-machine scheduling theory is that the sum of flowtimes (and hence the average flowtime) of all jobs is minimized by sequencing the jobs according to the SPT rule (Smith, 1956). As the release times of the jobs are assumed zero, the problem of minimizing the sum of flowtimes is equal to the problem of minimizing the sum of completion times. Note that the minimal flowtime problem is a special case of the common due date problem with $\alpha=\beta=0$ and $\theta=1$. Consequently, an optimal solution to the problem could be obtained by solving the assignment problem introduced above. However, the result of Smith (1956) still holds if the learning effect is taken into account (see the Theorem) and sequencing the jobs according to the SPT rule is easier than solving an assignment problem.

Theorem. *Incorporating learning effects into the processing of jobs according to Eq. (1), the sum of flowtimes* $\sum_{i=1}^{n} C_i$ *is minimized by the sequence* $(1, 2, \ldots, n)$, *i.e. the SPT order.*

Proof. The proof is established by the standard pairwise job interchange argument. Assume $p_i \leqslant p_j$ and job j is scheduled directly before job i in the rth position in a sequence, see Fig. 2(a). Let B and A be the sum of the flowtimes of the jobs scheduled before and after the jobs j and i, respectively and let C_{ji} be the overall objective function value. If interchanging the jobs j and i, as depicted in Fig. 2(b), does not increase the sum of its flowtimes, the value of A does not increase, either. Let C_{ij} be the sum of flowtimes yielded by the new sequence and let x be the completion time of the job occupying the $(r-1)$th position. Then

$$C_{ji} = B + (x + p_j r^a) + (x + p_j r^a + p_i (r+1)^a) + A$$

and

$$C_{ij} \leqslant B + (x + p_i r^a) + (x + p_i r^a + p_j (r+1)^a) + A.$$

Hence,

$$C_{ji} - C_{ij} \geqslant (p_j - p_i)(2r^a - (r+1)^a) \geqslant 0$$

for $p_i \leqslant p_j$. Note that $(2r^a - (r+1)^a)$ is non-negative, because $(r+1)^a < r^a$. Repeating this interchange argument for all jobs not sequenced according to the SPT order yields the theorem.

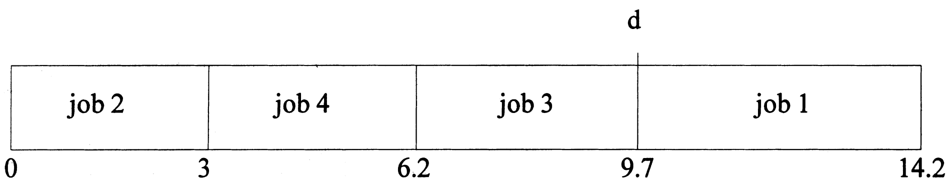

Fig. 1. The optimal schedule for Example 1.

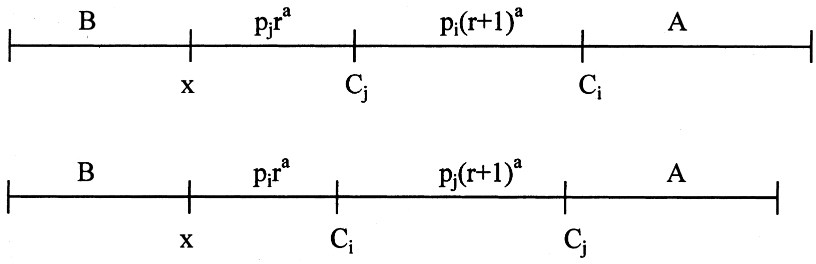

Fig. 2. A pairwise interchange of adjacent jobs.

Hence, for the minimal flowtime problem an optimal solution can be obtained by a sorting algorithm, thus taking O(n log n) time. The result of the Theorem is again demonstrated in the following example.

Example 1 (*continued*). Sequencing the jobs according to the SPT rule, (2, 4, 3, 1), yields the minimal sum of flowtimes with an objective function value of 33.1, see Table 2. The optimal schedule is depicted in Fig. 1.

5. Conclusions

It is shown in this paper that the single-machine scheduling problem with the consideration of learning effects remains polynomially solvable for two objectives, namely minimizing the deviation from a common due date and minimizing the sum of flowtimes.

Acknowledgements

I wish to thank Prof. Hermann Jahnke, University of Bielefeld, for extensive discussions and suggestions. I also thank two anonymous referees for their helpful comments on an earlier version of the article.

References

Alidaee, B., Ahmadian, A., 1993. Two parallel machine sequencing problems involving controllable job processing times. European Journal of Operational Research 70, 335–341.

Baker, K.R., 1974. Introduction to Sequencing and Scheduling. Wiley, New York.

Baker, K.R., Scudder, G.D., 1990. Sequencing with earliness and tardiness penalties: A review. Operations Research 38, 22–36.

Biskup, D., Cheng, T.C.E., 1997a. Single-machine scheduling with controllable processing times and earliness, tardiness and completion time penalties. Engineering Optimization, to appear.

Biskup, D., Cheng, T.C.E., 1997b. Multiple-machine scheduling with earliness, tardiness and completion time penalties. Computers & Operations Research, to appear.

Cheng, T.C.E., Chen, Z.L., Li, C.-L., 1996a. Parallel-machine scheduling with controllable processing times. IIE Transactions 28, 177–180.

Cheng, T.C.E., Oguz, C., Qi, X.D., 1996b. Due-date assignment and single machine scheduling with compressible processing times. International Journal of Production Economics 43, 29–35.

Higgins, P., Le Roy, P., Tierney, L., 1996. Manufacturing Planning and Control – Beyond MRP II. Chapman & Hall, London.

Kanet, J.J., 1981. Minimizing the average deviation of job completion times about a common due date. Naval Research Logistics Quarterly 28, 643–651.

Li, C.-L., Cheng, T.C.E., 1994. An economic production quantity model with learning and forgetting considerations. Production and Operations Management 3, 118–132.

Li, C.-L., Sewell, E.C., Cheng, T.C.E., 1995. Scheduling to minimize release-time resource consumption and tardiness penalties. Naval Research Logistics 42, S.949–966.

Nadler, G., Smith, W.D., 1963. Manufacturing progress functions for types of processes. International Journal of Production Research 2, 115–135.

Panwalkar, S.S., Dudek, R.A., Smith, M.L., 1973. Sequencing research and the industrial scheduling problem. In: Elmaghraby, S.E. (Ed.), Symposium on the Theory of Scheduling and its Applications. Springer, Berlin, pp. 29–38.

Panwalkar, S.S., Smith, M.L., Seidmann, A., 1982. Common due date assignment to minimize total penalty for the one machine scheduling problem. Operations Research 30, S.391–399.

Panwalkar, S.S., Rajagopalan, R., 1992. Single-machine sequencing with controllable processing times. European Journal of Operational Research 59, 298–302.

Papadimitriou, C.H., Steiglitz, K., 1982. Combinatorial Optimization: Algorithms and Complexity. Prentice-Hall, Englewood Cliffs, NJ.

Schmenner, R.W., 1988. The merit of making things fast. Sloan Management Review 30, 11–17.

Smith, W.E., 1956. Various optimizers for single-stage production. Naval Research Logistics Quarterly 3, 59–66.

Vasilev, S.H., Foote, B.L., 1997. On minimizing resource consumption with constraints on the makespan and the total completion time. European Journal of Operational Research 96, 612–621.

Vickson, R.G., 1980a. Choosing the job sequence and processing times to minimize total processing plus flow cost on a single machine. Operations Research 28, 1155–1167.

Vickson, R.G., 1980b. Two single machine sequencing problems involving controllable job processing times. AIIE Transactions 12, 258–262.

Wright, T.P., 1936. Factors affecting the cost of airplanes. Journal of Aeronautical Sciences 3, 122–128.

Yelle, L.E., 1979. The learning curve: historical review and comprehensive survey. Decision Science 10, 302–328.