Common due date scheduling with autonomous and induced learning

Dirk Biskup

Dirk Simons

Department of Business Administration and Economics
University of Bielefeld
PO-Box 10 01 31
33501 Bielefeld, Germany
Email: dsimons@wiwi.uni-bielefeld.de

Abstract
Yet, the few existing models considering learning effects in scheduling concentrate on learning-by-doing (autonomous learning). But recent contributions to the literature on learning in manufacturing organizations emphasize the important impact of proactive investments in technological knowledge on the learning rate (induced learning). In the present paper, we focus on a scheduling problem where the processing times decrease according to a learning rate which can be influenced by an initial cost-inducing investment. Thus we integrate into our model both aspects of learning -autonomous and induced- and thereby highlight the management's responsibility to invest in technological knowledge enhancement. We are able to derive some structural properties of the problem and present a polynomially bounded solution procedure which solves the problem to optimality by using these properties. The optimal solution of the scheduling problem contains -of course- information on the optimal level of proactive investments in learning.

1. Introduction and literature review

In the light of rapid technological progress, shortening life cycles and an increasing diversity of products, see, for example, Higgins et al. (1996), the importance of learning as a competitive advantage becomes self-evident. A prominent example is the semiconductor-industry where according to Webb (1994) the "dramatic improvement in efficiency accounts for the anticipation of price cuts of 10 to 30 percent each year as products mature." While the older part of the literature concentrates on learning-by-doing (autonomous learning), recent studies emphasize the importance of proactive investments in technological knowledge (induced learning) in order to increase learning effects. Thus, the distinction between autonomous and induced learning and the analysis of their interrelation seems to be crucial for understanding improvements in the factor productivity of labour. Consequently, we structure the introduction as follows. Firstly, some remarks on autonomous learning are made, followed by a discussion on induced learning. As both kinds of learning occur simultaneously and intensify each other, we afterwards introduce our model showing how autonomous and induced learning interact in the context of scheduling.

A decline in processing times usually can be observed when the same task is performed repeatedly. This means that the time needed to produce one product unit and thus the marginal and average costs per unit decrease in the cumulative production quantity. One of the pioneers discovering and describing this effect in a manufacturing setting was Wright (1936). Nevertheless, in psychology this phenomenon was known before, see Thurstone (1919, pp. 26), and an according theory has been developed there, see Mazur and Hastie (1978). Learning effects have received considerable attention in economics and management science, see the references given by Hatch and Mowery (1998, p. 1642). Among others the existence of learning effects was verified for different branches like machine building, the

production of music instruments or vehicle construction and for different divisions like servicing, construction or welding shops, see Hirsch (1952 and 1956), Conway and Schultz (1959), Nadler and Smith (1963), Hirschmann (1964) and Baloff (1971). In the recent literature the factor productivity's increase caused by a quality improvement of labour due to repetitions is called autonomous learning or learning-by-doing, see Ittner et al. (2001, p. 563), Lapré and van Wassenhove (2001, p. 1312) or Lapré et al. (2000, p. 598).

Frequently, autonomous learning is formalized by using the power form, $\tau(q) = \tau(1)q^a$, with τ symbolising the required processing time for the last unit produced given the cumulative production quantity q; $\tau(1)$ is the processing time of the first unit and a is the learning index, see e.g. Zangwill and Kantor (1998, p. 912). Despite its general acceptance several well known shortcomings are inherent in this traditional functional form, see Lapré et al. (2000, p. 598):

- It assumes a constant learning rate over the whole product's live. Prominent refinements relaxing this assumption are learning curves with an initial downward concavity, see Conway and Schultz (1959), curves allowing for the plateau effect, see Venezia (1985, p. 197), or S-shaped learning curves, see e.g. Cochran (1960, pp. 323). Additionally, synergy effects obtained from experiences with related products have to be considered as well, see Ghemawat (1985) or Day and Montgomery (1983). Nevertheless, all these refinements have in common that the learning rate is externally given and cannot be controlled.

- The power form suggests the decreasing costs per unit to be an automatism. However, it does not provide management with hints how to reduce costs beyond cost reductions resulting from learning-by-doing, see Zangwill and Kantor (1998, p. 910).

- The widely varying learning rates within industries or even plants indicate that the personnel's attitude towards learning has an impact on the learning rate as well, see

Sterman (1994, p. 293). The same is true for the personnel's knowledge diversity, see Lapré and van Wassenhove (2001, p. 1323). These facts are not reproduced adequately by an exogenously given learning rate.

Different from the autonomous learning, which is a by-product of repeating activities, induced learning requires management arrangements. The initial investment -as modelled in this paper- can be interpreted as induced learning via an investment in the creation of technological knowledge or knowledge transfer. The importance of this kind of learning has recently been highlighted by Ittner et al. (2001), Lapré and van Wassenhove (2001), Lapré et al. (2000), Hatch and Mowery (1998) and Mukherjee et al. (1998). Further, quality and design engineering influence the learning rate, too, see Ittner et al. (2001, p. 564); their influence on the learning rate and hence on the costs is -according to Sinclair et al. (2000) and Hatch and Mowery (1998, p. 1461)- much higher than the impact of the cumulative production quantities. Induced learning can be stimulated by means of employee's training, job-related instructions, handbooks or supervisors, etc. Additionally, product failures can be an important source of learning, see Schonberger (1982).

Generally, the definitions of autonomous and induced learning are not clear cut. While Hax and Majluf (1982) consider changes in the work-process separately from autonomous learning, Day and Montgomery (1983) subsume all effects causing a higher work-efficiency under this terminology. A reason for these different definitions might be that investments in learning (= induced learning) should also speed up the autonomous learning, clarifying that the two kinds of learning are interdependent as stated above. In the following we will refer to autonomous learning when learning is caused by repetition, whereas induced learning means every management activity in process design or controlled training.

For the short-term production planning, the role of learning can be inferred from the literature review given by Li and Cheng (1994). Although scheduling is one of the main tasks in short-term production planning, the effects of learning in this context have been considered by Biskup (1999) for the first time. He assumes the learning rate to be constant, i.e. he regards autonomous learning solely. With p_i, i = 1, …, n, being the normal processing time (i.e. processing time without any learning effects) of job i, Biskup (1999) assumes for a single-machine scheduling environment the following form of autonomous learning,

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

where p_{ir} is the processing time of job i if it is scheduled in position r and a = ld slr ≤ 0 is the learning index, given as the logarithm of the (standard) learning rate, slr, to the base 2. With respect to Eq. (1) the time needed to perform an operation decreases by the number of repetitions (Nadler and Smith, 1963 or Yelle, 1979), meaning that learning is primarily based on repetitions of time independent work-steps like set-ups, controlling and operating machines, reading data, etc. In this context Biskup (1999) was able to show for a single machine scheduling that the common due date problem with different (but not job-individual) penalties for earliness, tardiness and the completion time and the flowtime problem (minimizing the sum of flowtimes) can be solved in polynomial time. In the same context Mosheiov (2001a) showed that many well-known solutions for single-machine scheduling like the EDD (earliest due date) rule for minimizing maximum lateness, the WSPT (weighted smallest processing time) rule for minimizing the sum of weighted completion times or Moore's algorithm for minimizing the number of tardy jobs are not valid under learning assumptions. Furthermore, Mosheiov (2001a) demonstrated that two single-machine bi-criterial scheduling problems can be solved in polynomial time. Mosheiov (2001b) solved the flowtime minimization problem for the case that parallel identical machines exist.

The target of our paper is to model the prerequisites and effects of autonomous and induced learning in a single-machine scheduling environment and -in a second step- to find an optimal solution for this scheduling problem including an optimal level of induced learning. For achieving this goal, the remainder of the paper is organized as follows: In the second section, we present the model formulation. In the third section some structural properties of the objective function and of optimal schedules are proofed, which enable us to formulate a polynomially bounded solution procedure in the fourth section. The paper concludes with extensions of the basic model.

2. Model formulation

We have in mind a scheduling environment, where the effects of autonomous learning arise because similar jobs are produced one after the other on a single machine. The jobs have different normal processing times due to (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 set-ups 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 speed. Furthermore, we assume that the management is able to speed up this autonomous learning by an initial investment in learning. For modelling these effects of induced learning, we apply an improved version of the power form of Eq. (1).

We assume that there are n jobs available at time zero, which have to be processed on a single machine. Each job possesses a normal processing time p_i, i = 1, …, n, and the jobs are indexed according to the longest (normal) processing time (LPT) rule, i.e. $p_1 \geq p_2 \geq ... \geq p_n$. The normal processing time of a job is incurred solely if the job is scheduled first in a

sequence. Due to the learning effect the processing times of the following jobs are smaller than their normal processing times. As described in the introduction we assume that the learning can be controlled: If no learning effort takes place, the standard learning rate, slr, occurs and no further costs are incurred. But it might be advantageous to speed up the learning (i.e. to reduce the learning rate) and thus to stronger reduce the processing times of the later jobs. Decreasing the learning rate down to the new learning rate, nlr, is possible with respect to increasing learning costs. Let x be the percentage reduction of the standard learning rate and thus $nlr = (1 - x)slr$, then $k(x)$ are the costs which are incurred by the learning efforts. The cost function $k(x)$ is assumed to be weakly convex, i.e. the marginal costs of reducing the learning rate are non-decreasing. Furthermore $k(x)$ is assumed to be zero for $x = 0$. The maximal reduction of the learning rate is denoted by x_{max}.

The goal facing a common due date problem is to jointly minimize the sum of earliness and tardiness penalties, see Kanet (1981) or, for an introduction, Gordon et al. (2002) or Baker and Scudder (1990). An unrestricted common due date d is given, i.e. d is a decision variable or a fixed constant for which $d \geq \sum_{i=1}^{n} p_i$ holds. Furthermore C_i, $E_i = \max\{0, d - C_i\}$ and $T_i = \max\{0, C_i - d\}$ are the completion time, earliness and tardiness of job i, $i = 1, ..., n$, respectively. Then the general objective is to find a schedule π and a level of investments in learning, x, which jointly minimize

$$\text{KL: } f(\pi, x) = \sum_{i=1}^{n} \left(E_i + T_i \right) + k(x) \tag{2}$$

subject to

$$p_{ir} = p_i r^{\text{ld nlr}} \tag{3}$$

$$nlr = (1 - x)slr \tag{4}$$

$$0 \leq x \leq x_{max} < 1 \tag{5}$$

6

$$k(x) \text{ is a (weak) convex function in } x \qquad (6)$$

The problem is called KL (Kanet's problem with Learning) in the following. Our goal was to clearly model the cost induced learning in Eq. (2) to (6). Obviously this is not a complete model formulation, as the scheduling constraints are missing. They are well-known and can be found, for example, in a more general context in Biskup and Feldmann (2001).

Note that we distinguish between autonomous and induced learning and thereby highlight the management's responsibility for the learning speed, i.e. learning is subject to managerial discretion and control. Thus we are able to account for different learning rates within one plant. Further, modelling induced learning as cost-inducing investment clarifies that resources have to be dedicated to knowledge acquiring.

The reduction of processing times according to the job's position in the production sequence means that the influence of the total production quantity is rather small. The learning takes place after a job is finished, meaning that a concept similar to learning cycles as introduced by Zangwill and Kantor (1998, p. 911) is applied. After each job -you might think of it as a lot of identical units- mistakes are analysed and an improvement occurs when the next job is processed. Finally by modelling the learning rate as a function of the initial investment, the conjecture of Lapré et al. (2000, p. 608) that only learning projects acquiring both know-why and know-how lead to sustainable learning effects has been taken into consideration.

There exist several practical situations in which the problem KL occurs. Consider for example a single customer who places a large order over -let us say- twenty very special and somehow different machines. These machines shall be ready in half a year, because the

customer plans to replace his old machines then. Now the producing firm's management has to decide to which level the workers will be trained and in which order the machines will be produced. Machines which are finished too early tie up capital and machines which are finished too late dissatisfy the customer and might cause –for example- fines. Another example is that of a bulk delivery. If different customers from abroad -let us say US- order a number of goods, which are produced in Germany and if these goods are simultaneously dispatched (for example by boat), then the problem KL occurs. Again the management faces the decision problem to which level the workforce should be trained and in which order the goods should be produced to meet the bulk delivery due date. The goods which are late must be dispatched individually to higher costs and an early completion ties up capital.

3. Analysis and structural properties of optimal solutions

The learning effect of job i in position r only depends on the particular learning rate and the number of jobs processed prior to job i, i.e. the position of job i. The processing times of the jobs occupying the positions 1, 2, ..., r − 1 do not have any influence. Hence the learning effect for the jobs can be described on a positional basis independently of the particular processing times. For the job scheduled first no learning takes place, the positional weight is $1^{ld\ nlr}$. For the job on position two its normal processing time is multiplied with $2^{ld\ nlr}$. The weight for the job on the third position is $3^{ld\ nlr}$, etc. Hence the positional weights with respect to learning are $\gamma_r(\text{Learning}) = r^{ld\ nlr}$, $r = 1, \ldots, n$. Furthermore it is well-known that without learning the positional weights of Kanet's problem are $\gamma_r(\text{Kanet}) = \min\{(r - 1), (n - r + 1)\}$. Combining these results the objective function (2) can be rewritten as

$$f(\pi, x) = \sum_{r=1}^{n} \gamma_r p_{[r]} + k(x) \tag{7}$$

with $\gamma_r = \min\{(r - 1)r^{ld\ nlr}, (n - r + 1)r^{ld\ nlr}\}$ being the positional weight under learning which arises if a job occupies the r-th position. [r] indicates the job scheduled in the r-th position of the sequence.

Lemma 1: For a given x, respectively a given learning rate, KL can be solved by calculating the weights γ_r of Eq. (7) and matching the job with the longest processing time to the position with the smallest weight, the second longest job to the position with the second smallest weight etc.

Proof: Trivial from argumentation above.

Note that with respect to Lemma 1 the common due date problems tackled by Biskup (1999) and Mosheiov (2001a) could be solved much easier. The formulation of an assignment problem is not necessary, the particular problems could be solved by defining the positional weights first and afterwards applying a simple matching algorithm as presented in Lemma 1.

Lemma 2: For a given x, respectively a given learning rate, an optimal schedule for KL without idle times exists, in which the b-th job is completed at the due date where b is the smallest integer greater than or equal to n/2.

Proof: See Kanet (1981) or Panwalkar et al. (1982). Note that the number of tardy and non-tardy jobs for Kanet's problem is independent of the job processing times. As learning solely influences the processing times, these results can be transferred directly to KL for every value of x.

Lemma 2 describes two typical and useful properties for common due date scheduling. However, the well-known V-shaped property for optimal schedules –non-tardy jobs are

9

sequenced by decreasing processing times while tardy jobs are sequenced by increasing processing times– does not necessarily hold for KL.

Lemma 3: Instances of KL exist where non of the optimal schedules is V-shaped.

Proof: For n = 10 jobs and a learning rate nlr = 0.35 (a = −1.515) the following weights are valid: γ = (0 0.35 0.38 0.37 0.35 0.33 0.21 0.13 0.07 0.03). As the number of non-tardy jobs is n/2 = 5 (Lemma 2), the optimal schedule is not V-shaped: The shortest job is scheduled at position 3, the second shortest job at position 4, the third shortest job at position 2 or 5, etc.

Lemma 4: The later a job is sequenced the stronger it is effected by learning.

Proof: Obvious from the definition of learning and of γ_r(Learning).

From a theoretical point of view a consequence of Lemma 4 is that reducing the learning rate more and more leads to the final sequence := [1, n, n − 1, n − 2, ..., 2], which is always optimal for KL presupposed the costs k(x) are neglected. A further reduction of the learning rate does not cause any changes in this final sequence. For example with n = 10 and nlr = 0.3 the positional weights are γ = (0 0.300 0.297 0.27 0.244 0.223 0.136 0.081 0.044 0.018). The longest job is scheduled first, as its positional weight γ_1 is (always) zero. Furthermore it is advantageous to schedule the second longest job at the last, the third longest job at the penultimate position etc.

Define $h(x) := g\left(S^*\left(\underbrace{(1-x)}_{nlr} \, slr \right) \right) = \sum_{r=1}^{n} \gamma_r p_{[r]}$, where S* is the optimal schedule for a given x.

$h(x)$ is defined on the interval $[0; x_{max}]$.

Lemma 5: h(x) is monotone decreasing in x.

Proof: A formal proof is given in the appendix.

To gain access to the inherent structure of the problem, the following discussion of Lemma 5 is helpful: If no learning takes place (i.e. nlr = slr = 1, x = 0), the jobs are assigned to the positions as in Kanet's proble m, because the weights γ_r(Kanet) apply: The longest job is scheduled to the first position, the second longest job to the second or last position etc. Now, let us analyse the problem exemplarily: We start with nlr = 1 and increase x in (marginal) small steps. Because of Lemma 4 the positional weights do not alter proportional, but the weights of the rear positions of the sequence participate stronger from the learning than that of the prior positions. As soon as nlr is smaller than 1 assigning the second largest job alternatively to the second or last position is not valid any more. Now it is strictly advantageous to assign the second largest job to the last position of the sequence and the third largest job to the second position (again due to Lemma 4): The sequence [1, 3, 5, …, 6, 4, 2] is optimal. This sequence remains optimal –while decreasing the learning rate– until two positional weights, for example that of the second and penultimate position, i.e. $2^{\text{ld nlr}}$ and (n – (n – 1) + 1)$(n – 1)^{\text{ld nlr}}$, are equal. Let us define the underlying value of x to be a critical one, i.e. $\hat{x}$. For every $\hat{x}$ two different sequences are optimal, because two positional weights are equal and the jobs assigned to the corresponding positions can be interchanged without affecting the objective function value. If the learning rate is decreased any further, it is advantageous to interchange the jobs previously occupying these particular positions. Continuing the example the sequence [1, 4, 5,…, 6, 3, 2] would be optimal now. Please note that it not necessarily have to be the second and the penultimate jobs which are interchanged first - this depends on the overall number of jobs. By increasing x, i.e. decreasing the learning

rate, more and more this interchange procedure continues at most until the final sequence is reached.

From the argumentation above two aspects should have become evident: Firstly, the objective function value decreases with an increasing x, i.e. a decreasing learning rate. Secondly, there are no jump discontinuities, as every job interchange in the optimal sequence takes place if (and only if) two weights are equal. Hence the objective function is monotone decreasing in x respectively in a decreasing learning rate.

Lemma 6: $h(x)$ is piecewise convex.

Proof: Again, a formal proof can be found in the appendix.

Similar to Lemma 5, we give a discussion of Lemma 6. We start with nlr = 1 and decrease the learning rate by increasing x continuously. As long as a particular sequence is optimal, i.e. as long as no interchange of jobs is necessary, $h(.)$ is convex in the particular interval. This becomes evident from the fact that for all positions of the sequence the contribution to the objective function value is calculated by multiplying $ct_r = p_{[r]} \cdot \gamma_r(\text{Kanet})$ with $r^{ld\ nlr}$, $r = 1, ..,$ n. $ct_r \cdot r^{ld\ nlr}$ is obviously convex in nlr. The sum of convex functions results in a convex function again. This argumentation is valid as long as the underlying sequence is optimal. From a certain value of nlr on a different sequence -obtained by interchanging two jobs- becomes optimal. For this sequence the same argumentation applies etc.

$h(x)$ is only piecewise convex (and not convex), because after interchanging two jobs the factors $p_{[r]}$ are different at two positions compared to that of the former sequence. This results in different slopes of the curve $h(x)$ for different optimal sequences.

4. Solution procedure

From Lemma 6 it is known that h(x) is not convex but piecewise convex. We exploit this property in our solution procedure by searching for optimal x's within each interval. By increasing x there are critical values $\hat{x}$ at which two different sequences are optimal. A convex interval of h(x) is bordered by two adjacent critical $\hat{x}$'s. Note that -from the definition of the intervals- the underlying optimal sequence does not change within an interval. For finding the global optimum of (2) it is necessary to analyse all intervals of h(x). As k(x) is considered to be convex, too, for each interval of h(x) a minimum objective function value for KL exists. This might be a solution at the border. Furthermore, from Lemma 1 it is obvious that for a given x or a given learning rate respectively the optimal sequence and thus the optimal objective function value can be found in polynomial time. Consequently, the following solution procedure can be applied:

1. Identify the intervals $[0; \hat{x}_1], [\hat{x}_1; \hat{x}_2], ..., [\hat{x}_m; x_{max}]$: Calculate $\hat{x}$ for each pair of positional weights. This is done by equalling two weights of γ_r and isolating for $\hat{x}$. Order the $\hat{x}$ by increasing values, i.e. $\hat{x}_1 \le \hat{x}_2 \le \hat{x}_3 \le ... \le \hat{x}_m$. As n(n - 1)/2 pairs of positional weights have to be regarded, the maximal number of intervals equals n(n – 1)/2. The real number of m + 1 intervals might be somewhat smaller either as some $\hat{x}$ can be larger than x_{max} or some of the $\hat{x}$ are identical.

2. Let $\kappa = 1$ and $f_{best} = n \sum_{i=1}^{n} p_i$.

3. Analyse the interval κ:

   - Identify the optimal sequence S_κ by Lemma 1.

- Find the minimal objective function value f_κ for $f(\pi, x)$ at x_κ. This can be done by simple functional analysis, for example via the first derivative, as S_κ does not change and hence $f(\pi, x)$ is convex in the interval κ.

- If $f_\kappa < f_{best}$, then $f_{best} := f_\kappa$, $S_{best} := S_\kappa$ and $x_{best} := x_\kappa$.

4. If $\kappa > m$, terminate the search procedure.

5. Increase κ by one and continue with step 3.

Example: Let $n = 10$ with $p_i = 11 - i$, $slr = 1$ and $k(x) = 250x^2$.

The first three intervals -rounded five positions after the comma- are [0; 0.24914], [0.24915; 0.27343], [0.27344; 0.29975]. For example, the border of the first interval is obtained by solving $2 \cdot 3^{ld(1-\hat{x})} = 3 \cdot 8^{ld(1-\hat{x})}$ to $\hat{x} = 0.249143$ (which is the smallest of all $\hat{x}$). A consequence from this fact is that the jobs at the positions 3 and 8, i.e. jobs number 5 and 6, interchange first. The optimal solution obtained by the solution procedure lies in the third interval with $x = 0.283$ yielding an objective function value of 62.8281. The optimal sequence is $S^* = (1, 4, 6, 7, 9, 10, 8, 5, 3, 2)$. The due date is set so that the fifth job is completed exactly in d. If a unrestrictive large due date was given, the first job is started so that again the fifth job is completed exactly in d.

5. Extensions and summary

We demonstrated the effects of autonomous and induced learning on the problem of Kanet, as this is well-known and somehow simple but, nevertheless, a very interesting scheduling problem. The solution procedure presented is not restricted to the problem of Kanet. It can be extended easily to many more common due date problems. Namely to all common due date problems with an objective function consisting of positional weights, i.e. which can be

reformulated as in Eq. (7). These are unrestricted single-machine common due date problems with

- ♣ different, but not job-individual earliness and tardiness penalties α, β:

$$f_I(\pi, x) = \sum_{i=1}^{n}\left(\alpha E_i + \beta T_i\right) + k(x) = \sum_{r=1}^{n}\gamma_{r,I}p_{[r]} + k(x)$$

with $\gamma_{r,I} = \min\{\alpha(r-1)r^{ld\ nlr}, \beta(n-r+1)r^{ld\ nlr}\}$.

- ♣ a penalty for assigning a late common due date γ:

$$f_{II}(\pi, x) = \sum_{i=1}^{n}\left(\alpha E_i + \beta T_i + \gamma d\right) + k(x) = \sum_{r=1}^{n}\gamma_{r,II}p_{[r]} + k(x)$$

with $\gamma_{r,II} = \min\{n\gamma + \alpha(r-1)r^{ld\ nlr}, n\gamma + \beta(n-r+1)r^{ld\ nlr}\}$.

- ♣ a penalty on the completion time of the jobs θ:

$$f_{III}(\pi, x) = \sum_{i=1}^{n}\left(\alpha E_i + \beta T_i + \theta C_i\right) + k(x) = \sum_{r=1}^{n}\gamma_{r,III}p_{[r]} + k(x)$$

with $\gamma_{r,III} = \min\{(n-r+1)\theta + \alpha(r-1)r^{ld\ nlr}, (n-r+1)\theta + \beta(n-r+1)r^{ld\ nlr}\}$.

Baker and Scudder (1990) present an excellent introduction to these kind of common due date problems.

We have been able to state a formulation of the common due date scheduling problem with autonomous and induced learning effects. Furthermore, we could prove some structural properties enabling us to develop a polynomial bounded solution procedure. The presented procedure is not restricted to the Problem of Kanet but can be applied to all common due date problems with an objective function basing on positional weights. From a managerial perspective it becomes obvious that a global optimum even in a production planning environment typically can only be achieved if learning by doing is accompanied by management activities supporting induced learning. To the best of our knowledge within this

paper autonomous and induced learning have been considered simultaneously for the first time in the context of scheduling.

References

Baker, K. R. and G. D. Scudder (1990), Sequencing with earliness and tardiness penalties: a review, *Operations Research* 38, 22-36.

Baloff, N. (1971), Extension of the learning curve – some empirical results, *Operational Research Quarterly* 22, 329-340.

Biskup, D. (1999), Single-machine scheduling with learning considerations, *European Journal of Operational Research* 115, 173-178.

Biskup, D., M. Feldmann (2001), Benchmarks for scheduling on a single machine against restrictive and unrestrictive common due dates, *Computers & Operations Research* 28, 787-801.

Cochran, E. B. (1960), New concepts of the learning curve, *The Journal of Industrial Engineering* 11, 317-327.

Conway, R. W. and A. Schultz (1959), The manufacturing progress function, *The Journal of Industrial Engineering* 10, 39-54.

Day, G. S. and D. B. Montgomery (1983), Diagnosing the experience curve, *Journal of Marketing* 47, 44-58.

Ghemawat P. (1985), Building strategy on the experience curve – a venerable management tool remains valuable – in the right circumstances, *Harvard Business Review* 63 II, 143-149.

Gordon, V., J-M Proth and C. Chu (2002), A survey of the state-of-the-art of common due date assignment and scheduling research, *European Journal of Operational Research* 139, 1-25.

Hatch, N. W. and D. C. Mowery (1998), Process innovation and learning by doing in semiconductor manufacturing, *Management Science* 44, 1461-1477.

Hax, A. C. and N. S. Majluf (1982), Competitive cost dynamics: The experience curve, *Interfaces* 12, 50-61

Higgins P., P. Le Roy, L. Tierney (1996), *Manufacturing, planning and control – beyond MRP II*, Chapman & Hall, London.

Hirsch, W. Z. (1952), Manufacturing progress functions, *The Review of Economics and Statistics* 34, 143-155.

Hirsch, W. Z. (1956), Firm progress ratios, *Econometrica* 24, 136-143.

Hirschmann, W. B. (1964), Profit from the learning curve, *Harvard Business Review*, 125-139

Ittner, C. D., V. Nagar and M. V. Rajan (2001), An empirical examination of dynamic quality-based learning models, *Management Science* 47, 563-578.

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

Lapré, M. A., A. S. Mukkherjee and L. N. Van Wassenhove (2000), Behind the learning curve: Linking learning activities to waste reduction, *Management Science* 46, 597-611.

Lapré, M. A. and L. N. Van Wassenhove (2001), Creating and transferring knowledge for productivity improvement in factories, *Management Science* 47, 1311-1325.

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

Mazur, J.E., R. Hastie (1978), Learning as accumulations: A reexamination of the learning curve, *Psychological Bulletin* 85, 1256-1274.

Mosheiov, G. (2001a), Scheduling problems with a learning effect, *European Journal of Operational Research* 132, 687-693.

Mosheiov, G. (2001b), Parallel machine scheduling with a learning effect, *Journal of the Operational Research Society* 52, 1-5.

Mukherjee, A. S., M. A. Lapré and L. N. van Wassenhove (1998): Knowledge driven quality improvement, *Management Science* 32, 948-962.

Nadler, G. and W. D. Smith (1963): Manufacturing progress functions for types of processes, *International Journal of Production Research* 2, 115-135.

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

Schonberger, R. J. (1982), Some observations on the advantages and implementation issues of just-in-time production systems, *Journal of Operations Management* 3, 1-11.

Sinclair, G., K. Steven and C. Wessley (2000), What's experience got to do with it? Sources of cost reduction in a large specialty chemicals producer, *Management Science* 46, 28-45.

Sterman, J.D. (1994), Learning in and about complex systems, *System Dynamics Review* 10, 291-330.

Thurstone, L. L. (1919), The learning curve equation, *Psychological Monographs* 26, 1-51.

Tiel, J. van (1984), *Convex analysis*, Wiley, Chichester.

Venezia, I. (1985), On the statistical origins of the learning curve, *European Journal of Operational Research* 19, 191-200.

Webb, G.K., (1994) Integrated circuit (IC) pricing, *High Technology Management Research* 5 II, 247-260.

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.

Zangwill, W. I. and P. B. Kantor (1998), Toward a theory of continuous improvement and the learning curve, *Management Science* 44, 910-920.

Appendix:

In the following a formal proof of Lemma 5 and Lemma 6 is given:

Define $h(x) := g(S^*((1-x)slr)) = \sum_{r=1}^{n} \gamma_r p_{[r]}$, where S^* is the optimal schedule for a given x. $h(x)$ is defined on the interval $[0; x_{max}]$. This interval can be divided into certain subintervals $[0; \hat{x}_1], [\hat{x}_1; \hat{x}_2], ..., [\hat{x}_m; x_{max}]$, where $\hat{x}_1$, $1 = 1, ..., m$, indicates a critical x, which induces a change of the optimal schedule.

Proposition: $h(x)$ is piecewise convex on the interval $[0; x_{max}]$.

Proof: The proof consists of two parts. In the first part it will be shown that $h(x)$ is convex in each subinterval, where the optimal schedule does not alter with a variation of x. In the second step it will be shown that for any $\hat{x}$ the left derivative can be greater than the right derivative, contradicting the definition of convexity.

(i) Considering a subinterval, meaning the optimal schedule does not change, $h(x)$ can be written as:

$$h(x) = \sum_{r=1}^{n} \gamma_r p_{[r]} = \sum_{r=1}^{n} \min\{(r-1), (n-r+1)\} r^{ld[(1-x)slr]} p_{[r]} \qquad (A1)$$

Obviously, all the terms are positive. Further only $r^{ld[(1-x)slr]}$ depends on x. Hence the first-order derivative with respect to x is:

$$\frac{dh(x)}{dx} = \sum_{r=1}^{n} \min\{(r-1),(n-r+1)\} p_{[r]} r^{ld\,[(1-x)slr]} \ln(r)\left(-\frac{1}{1-x}ld(e)\right) < 0 \qquad (A2)$$

The second-order derivative is:

$$\frac{d^2h(x)}{dx^2} = \sum_{r=1}^{n} \min\{(r-1),(n-r+1)\} p_{[r]} \ln(r) r^{ld[(1-x)slr]}\left[ld(e)\left(\frac{1}{1-x}\right)^2(\ln(r)ld(e)-1)\right] \geq 0$$

$$(A3)$$

(A3) is strictly positive for n > 2 (and zero for n = 1 or n = 2). From (A3) it follows that $h(x)$ is monotone decreasing and convex in x as long as the sequence remains unchanged, completing the first part of the proof.

(ii) In the second part, it has to be shown that $h(x)$ might not be convex over $[0; x_{max}]$.

Theorem: Let $f : I \rightarrow \Re$ be convex. Then f has a right derivative, f'_{right}, and a left derivative, f'_{left}, at every point of the interior of the interval I, $int(I)$, and f'_{left} and f'_{right} are non-decreasing on $int(I)$. If $c \in int(I)$, we have $f'_{left}(c) \leq f'_{right}(c)$.

Proof: See van Tiel (1984, p. 4).

Considering two adjacent subintervals $[\hat{x}_{l-1};\hat{x}_l],[\hat{x}_l;\hat{x}_{l+1}]$ at least two sequences yield the same objective function value at $\hat{x}_l$. Let S_1^* be the sequence with job j scheduled on position t and job k scheduled on position q > t. Let S_2^* be the sequence with job k scheduled on position t before job j on position q. $h(\hat{x}_l)$ can be written as follows:

$$h(\hat{x}_l) = \sum_{r=1,r\neq q,t}^{n}\gamma_r p_{[r]} + \underbrace{(t-1)t^{ld[(1-x)slr]}}_{w}p_j + \underbrace{(n-q+1)q^{ld[(1-x)slr]}}_{y}p_k =$$
$$\sum_{r=1,r\neq j,k}^{n}\gamma_r p_{[r]} + \underbrace{(t-1)t^{ld[(1-x)slr]}}_{w}p_k + \underbrace{(n-q+1)q^{ld[(1-x)slr]}}_{y}p_j = h(\hat{x}_l)$$

$$(A4)$$

In the upper row the objective function value is computed given S_1^* and the second row is computed given S_2^*. Let $p_j > p_k$, meaning that we consider a longer job to become scheduled

later in accordance with Lemma 4. For (A4) to hold, the positional weights have to be equal, i.e. $w = y$.

Note that by changing the positions of jobs j and k, the contribution of all other jobs to the objective function value is constant. This means all the terms for $r \neq q, t$ remain unchanged and we can symbolize them by a constant ä, when considering the left and right hand side derivative:

$h'_{left}(\hat{x}_1) =$

$$\delta + p_j t^{ld[(1-x)slr]} \underbrace{(t-1)\left(-\frac{1}{1-x}ld(e)\right)\ln(t)}_{a}^{\;w} + p_k q^{ld[(1-x)slr]} \underbrace{(n-q+1)\left(-\frac{1}{1-x}ld(e)\right)\ln(q)}_{b}^{\;y} >$$

$$\delta + p_k t^{ld[(1-x)slr]} \underbrace{(t-1)\left(-\frac{1}{1-x}ld(e)\right)\ln(t)}_{a}^{\;w} + p_j q^{ld[(1-x)slr]} \underbrace{(n-q+1)\left(-\frac{1}{1-x}ld(e)\right)\ln(q)}_{b}^{\;y}$$

$$= h'_{right}(\hat{x}_1)$$

$$(A5)$$

In accordance with $p_j > p_k$ let $q > r$ hold. Because of $w = y$ and $\ln(r) < \ln(q)$, (A5) can be simplified to:

$$p_j a + p_k b > p_k a + p_j b \qquad (A6)$$

with $0 > a > b$.

Showing that (A6) is true completes the proof:

$$p_j a + p_k b > p_k a + p_j b$$
$$\Leftrightarrow a(p_j - p_k) > b(p_j - p_k) \qquad (A7)$$
$$\Leftrightarrow \frac{a}{b} < 1$$

The inequality changes because dividing by b means dividing by a negative term. Obviously (A7) and hence (A5) are true, contradicting the Theorem stated above. Hence $h(x)$ is piecewise convex on the interval $[0; x_{max}]$.