An FPTAS for Scheduling with Piecewise-Linear Nonmonotonic Convex Time-Dependent Processing Times and Job-Specific Agreeable Slopes

Helmut A. Sedding

Institute of Data Analysis and Process Design
ZHAW Zurich University of Applied Sciences, Winterthur, Switzerland

Keywords: Time-dependent scheduling, Piecewise-linear convex nonmonotonic processing times with agreeable ratios of basic processing time and slopes, FPTAS

1 Introduction

The time-dependent scheduling branch is concerned with processing times that are a function of job start time (Gawiejnowicz 2020a, 2020b). These effectively superimpose an additional layer of complexity compared to fixed processing times. For example, an interchange of adjacent jobs typically yields a change in their processing time. Additionally, this change affects all subsequent jobs' processing time. Thus, applying an adjacent job interchange argument is more involved. Hence, it can be challenging already to sequence a set of jobs without idle time on a single machine with the objective of minimizing the makespan $C_{\max}$. This problem is considered in this study for a processing time function that can attain a nonmonotonic convex shape. This shape distinguishes it from most existing literature, which considers monotonic convex shapes (Gawiejnowicz 2020a, 2020b).

Time-dependent effects of a job's start time can be additive, multiplicative, or combined (Strusevich and Rustogi 2017). The effect studied in this study is additive, extending the earliest work in this field in Shafransky (1978). Here, each job j has a *basic processing time* ℓ_j, and a *penalty function* f_j of start time t. They are added to yield processing time

$$p_j(t) = \ell_j + f_j(t). \tag{1}$$

This study considers the nonmonotonic piecewise-linear job-specific penalty function

$$f_j(t) = \max\{-a_j\,(t-\tau),\, b_j\,(t-\tau)\} \tag{2}$$

for a given common *ideal start time* τ and rational valued *slopes* $0 \leq a_j \leq 1$ and $b_j \geq 0$. The jobs are required to have *agreeable ratios of basic processing time and slopes*, i.e., there must exist a sequence all jobs that fulfills condition

$$\ell_i a_j \geq \ell_j a_i \quad \text{and} \quad \ell_i b_j \geq \ell_j b_i \quad \text{for any job } i \text{ sequenced before any job } j. \tag{3}$$

The main challenge of this problem is to decide which jobs shall start before τ, and which jobs shall start at or after τ. This decision is NP-hard already for uniform slopes $a = a_j$, $b = b_j$, as shown in Sedding (2020b, 2020c) by reduction from Even-Odd Partition.

In this study, a fully polynomial time approximation scheme (FPTAS) is given for this problem, which enables to sequence the jobs both quickly and with an error bound.

In practice, this enables fast reaction times in computationally sequencing a worker's tasks at the moving conveyor line, e.g., in car assembly. Here, each operation involves leaving the work point, walking along the line to a supply point, and return. The occurring walking time is, according to Sedding (2020a), adequately depicted by the studied penalty function (2). Individual slopes with agreeable ratios (3) reflect task-specific walking velocities that occur, e.g., when carrying additional weight.

2 Related problems

Existing time-dependent scheduling literature with an additive effect mostly considers penalty functions that are monotonic, hence either nondecreasing (often called *deterioration effect*) or nonincreasing (sometimes called *learning effect*). The main advantage of monotonic penalty functions is that their monotonic effect is recursive. Starting some job earlier does not increase its own completion time, and neither those of successors.

An interesting discovery is that it is possible for any shape of uniform ($f = f_j$) monotonic penalty functions to find an optimal sequence in polynomial time: by sorting the jobs with respect to their basic processing time (Melnikov and Shafransky 1979).

A well-known polynomial case with job-specific penalty functions has the non-increasing proportional-linear $f_j(t) = -a_j\, t$ with $0 \leq a_j \leq 1$ (hence, $b_j = 0$ and $\tau = 0$ in (2)). Here, any sequence that fulfills condition (3) is optimal (Ho, Leung and Wei 1993). The symmetric case is the non-decreasing proportional-linear $f_j(t) = b_j\, t$ with $b_j \geq 0$ (hence, $a_j = 0$ and $\tau = 0$ in (2)). Here, an optimal sequence fulfills condition

$$\ell_i a_j \leq \ell_j a_i \quad \text{and} \quad \ell_i b_j \leq \ell_j b_i \quad \text{for any job } i \text{ sequenced before any job } j \tag{4}$$

(Shafransky 1978, Gupta and Gupta 1988, Browne and Yechiali 1990, Gawiejnowicz and Pankowska 1995).

These results are the basis for the monotonic piecewise-linear case, expressed by penalty functions as in (2) but restricting slopes either to $a = a_j = 0$, or to $b = b_j = 0$. Then, the jobs need to be partitioned into two sides around τ, which is an NP-hard problem (Kononov 1997, Kubiak and van de Velde 1998, Cheng, Ding, Kovalyov, Bachman and Janiak 2003) that permits FPTASs (Kovalyov and Kubiak 1998, Kovalyov and Kubiak 2012, Cai, Cai and Zhu 1998, Woeginger 2000, Ji and Cheng 2007, Halman 2020).

The non-monotonic penalty function case in (2) with symmetric slopes $a_j = b_j < 1$ is covered by the model in Kononov (1998) for all-zero basic processing times $\ell_j = 0$, and solved by ordering the jobs non-decreasingly with respect to a_j. Nonnegative $\ell_j \geq 0$ are first considered in Sedding and Jaehn (2014) for uniform symmetric slopes $a = a_j = b_j < 1$. A similar model is studied in Jaehn and Sedding (2016), a much more general model in Kawase, Makino and Seimi (2018). In Sedding (2020b, 2020c), the uniform slopes $a = a_j$, $b = b_j$ case is shown to be NP-hard. This model is extended to agreeable slope ratios (3) in this study, which is expanded in Sedding (2020b).

3 Sorting criteria

In the given problem, optimal sequences exhibit a certain sort order for the jobs that complete before or at the ideal start time τ (they are denoted by partial sequence S_1), and another for the jobs starting at or after τ (denoted by S_2). Then, the jobs in S_1 and S_2 effectively have proportional-linear penalty functions. Hence, the respective sum of processing times in S_1 and S_2 is minimized if the sorting criteria as described in section 2 hold, i.e., if S_1 fulfills condition (3) and S_2 fulfills (4) (Sedding 2018a, 2018b).

The sorting criteria on both S_1 and S_2 have implications on the construction of the FPTAS. In the monotonic case, an arbitrary sorting is possible for either S_1 (if all $a_j = 0$), or S_2 (if all $b_j = 0$) because the processing times in one of them are not time-dependent. Assuming these unchanging processing times are integer, one can also assume an integral, pseudopolynomial sum of processing times. Note that the known FPTAS for these problems utilize both properties, which leaves them unsuitable for the studied problem.

At least, the sort criteria in S_1 and in S_2 are related as follows. A sequence for condition (3) exists, and it is found in polynomial time. With this, the jobs are *agreeably* *renumbered* such that (3) holds for the job sequence $1, 2, \ldots, n$. Then, it follows that there exists an optimal S_1 where the jobs are increasingly numbered, and an optimal S_2 where they are decreasingly numbered. Hence, S_1 and S_2 can be symmetrically sorted.

Please note that an exception to these sorting criteria might exist with a *straddler job* that starts before or at τ and completes at or after τ. In particular, such a job might not be the last job according to these criteria, i.e., the job number with number n.

4 Dynamic programming algorithm

The following dynamic programming algorithm solves the given problem exactly if a straddler job exists (if not, the instance corresponds to a proportional-linear penalty function case) and is already given. To choose it, the algorithm is repeatedly started with each of the jobs as a straddler job, then returning the best feasible schedule. In the following, straddler job χ has been chosen, the others are agreeably renumbered to $1, 2, \ldots, n$.

Then, the dynamic program consists of stages 1 to n. Each stage $j \in \{1, \ldots, n\}$ generates a set V_j of partial solutions. To generate this set, job j is inserted into all partial solutions of the preceding stage V_{j-1}, beginning with an empty solution in the first stage. Each partial solution represents two sequences S_1, S_2. Sequence S_1 represents the jobs to be completed before or at τ, and S_2 the jobs to be started at or after τ. A partial solution in V_{j-1} includes the jobs $1, \ldots, j-1$ and is encoded by a nonnegative real vector $[x, y, z]$ of

- x, which specifies sequence S_1's completion time,
- y, which specifies the proportional increase (i.e., the value of the partial derivative) of sequence S_2's completion time for increasing its start time, and
- z, which specifies sequence S_2's sum of processing times if it is started at τ.

The initial partial solution set is $V_0 = \{[0, 1, 0]\}$. There are two ways to add job j to a partial solution: either appending j to S_1 (if possible), or prepending j at S_2. In this way, condition (3) is always upheld for S_1, and (4) for S_2. From any vector $[x, y, z] \in V_{j-1}$, appending job j to S_1 is possible if $x + p_j(x) \le \tau$, and it adds vector $[x + p_j(x),\, y,\, z]$ to V_j. Prepending j to S_2 adds another vector $[x,\, y \cdot (1 + b_j),\, y \cdot \ell_j + z]$ to V_j.

After the final stage n, the straddler job χ is inserted between S_1 and S_2. Given a vector $[x, y, z] \in V_n$ of the final stage, let χ start at x. If the completion time $C_\chi = x + p_\chi(x)$ of the straddler job is less than τ, the vector is discarded. Otherwise, the makespan $C_{\max} = \tau + y \cdot (C_\chi - \tau) + z$ of the solution vector is obtained. The smallest $C_{\max}$ of all $[x, y, z] \in V_n$ is the optimum makespan value $C^*_{\max}$ (assuming χ is an optimal straddler job). The according job sequence can be reconstructed by traveling back the corresponding partial solutions.

5 A fully polynomial time approximation scheme

The dynamic program is turned into an FPTAS by trimming the states using exponentially growing value bins as described in Woeginger (2000). For a given maximum relative error $\varepsilon \in (0, 1]$, define $\Delta = 1 + \frac{\varepsilon}{2n}$, and function $h(x) = \Delta^{\lceil \log_\Delta x \rceil}$ for any positive real x, which satisfies $x/\Delta < h(x) \le x \cdot \Delta$. After each stage j, the set of partial solutions V_j is trimmed: for any disjoint pair of vectors $[x, y, z] \in V_j$, $[x', y', z'] \in V_j$ where $x \le x'$, $h(y) \le h(y')$, and $h(z) = h(z')$, the vector $[x', y', z']$ is discarded. It can be shown that the number of vectors in V_n remains polynomially bounded in input size and $1/\varepsilon$ by $\mathcal{O}(n^3 \cdot \log(1 + b_{\max}) \cdot (\log \max\{\ell_{\max}, 1/b_{\max}\} + n \log(1 + b_{\max}))/\varepsilon^2)$ for $\ell_{\max} = \max_j \ell_j$, $b_{\max} = \max_j b_j$; and a bounded minimum makespan $C^{\text{approx}}_{\max} \le C^*_{\max} \cdot (1 + \varepsilon)$ is achieved.

References

Browne, S. and Yechiali, U.: 1990, Scheduling deteriorating jobs on a single processor, *Operations Research* **38**(3), 495–498.

Cai, J.-Y., Cai, P. and Zhu, Y.: 1998, On a scheduling problem of time deteriorating jobs, *Journal of Complexity* **14**(2), 190–209.

Cheng, T. C. E., Ding, Q., Kovalyov, M. Y., Bachman, A. and Janiak, A.: 2003, Scheduling jobs with piecewise linear decreasing processing times, *Naval Research Logistics* **50**(6), 531–554.

Gawiejnowicz, S.: 2020a, *Models and algorithms of time-dependent scheduling*, Monographs in Theoretical Computer Science, second edn, Springer, Berlin, Heidelberg.

Gawiejnowicz, S.: 2020b, A review of four decades of time-dependent scheduling: main results, new topics, and open problems, *Journal of Scheduling* **23**(1), 3–47.

Gawiejnowicz, S. and Pankowska, L.: 1995, Scheduling jobs with varying processing times, *Information Processing Letters* **54**(3), 175–178.

Gupta, J. N. D. and Gupta, S. K.: 1988, Single facility scheduling with nonlinear processing times, *Computers & Industrial Engineering* **14**(4), 387–393.

Halman, N.: 2020, A technical note: fully polynomial time approximation schemes for minimizing the makespan of deteriorating jobs with nonlinear processing times, *Journal of Scheduling* **23**(6), 643–648.

Ho, K. I.-J., Leung, J. Y.-T. and Wei, W.-D.: 1993, Complexity of scheduling tasks with time-dependent execution times, *Information Processing Letters* **48**(6), 315–320.

Jaehn, F. and Sedding, H. A.: 2016, Scheduling with time-dependent discrepancy times, *Journal of Scheduling* **19**(6), 737–757.

Ji, M. and Cheng, T. C. E.: 2007, An FPTAS for scheduling jobs with piecewise linear decreasing processing times to minimize makespan, *Information Processing Letters* **102**(2-3), 41–47.

Kawase, Y., Makino, K. and Seimi, K.: 2018, Optimal composition ordering problems for piecewise linear functions, *Algorithmica* **80**(7), 2134–2159.

Kononov, A. V.: 1997, On schedules of a single machine jobs with processing times nonlinear in time, *Discrete Analysis and Operational Research* **391**, 109–122.

Kononov, A. V.: 1998, Problems in scheduling theory on a single machine with job durations proportional to an arbitrary function, *Diskretnyĭ Analiz i Issledovanie Operatsiĭ* **5**(3), 17–37.

Kovalyov, M. Y. and Kubiak, W.: 1998, A fully polynomial approximation scheme for minimizing makespan of deteriorating jobs, *Journal of Heuristics* **3**(4), 287–297.

Kovalyov, M. Y. and Kubiak, W.: 2012, A generic FPTAS for partition type optimisation problems, *International Journal of Planning and Scheduling* **1**(3), 209.

Kubiak, W. and van de Velde, S. L.: 1998, Scheduling deteriorating jobs to minimize makespan, *Naval Research Logistics* **45**(5), 511–523.

Melnikov, O. I. and Shafransky, Y. M.: 1979, Parametric problem in scheduling theory, *Cybernetics* **15**(3), 352–357.

Sedding, H. A.: 2018a, On the complexity of scheduling start time dependent asymmetric convex processing times, *Proceedings of the 16th International Conference on Project Management and Scheduling*, Università di Roma "Tor Vergata", Rome, Italy, pp. 209–212.

Sedding, H. A.: 2018b, Scheduling non-monotonous convex piecewise-linear time-dependent processing times, *2nd International Workshop on Dynamic Scheduling Problems*, Adam Mickiewicz University, Poznań, Poland, pp. 79–84.

Sedding, H. A.: 2020a, Line side placement for shorter assembly line worker paths, *IISE Transactions* **52**(2), 181–198.

Sedding, H. A.: 2020b, Scheduling jobs with a V-shaped time-dependent processing time, *Journal of Scheduling* **23**(6), 751–768.

Sedding, H. A.: 2020c, *Time-dependent path scheduling: Algorithmic minimization of walking time at the moving assembly line*, Springer Vieweg, Wiesbaden.

Sedding, H. A. and Jaehn, F.: 2014, Single machine scheduling with nonmonotonic piecewise linear time dependent processing times, *Proceedings of the 14th International Conference on Project Management and Scheduling*, TUM School of Management, Munich, Germany, pp. 222–225.

Shafransky, Y. M.: 1978, On optimal ordering in deterministic systems with tree-like partial serving order, *Proceedings of the Academy of Sciences of Belarus, Physics&Mathematics* **1978**(2), 120.

Strusevich, V. A. and Rustogi, K.: 2017, *Scheduling with time-changing effects and rate-modifying activities*, Springer, Cham.

Woeginger, G. J.: 2000, When does a dynamic programming formulation guarantee the existence of a fully polynomial time approximation scheme (FPTAS)?, *INFORMS Journal on Computing* **12**(1), 57–74.