Péter Györgyi

Approximation and exact methods for machine scheduling problems with non-renewable resources

Summary of the Ph.D. thesis

Supervisor: Tamás Kis, Ph.D.

senior research fellow, Institute for Computer Science and Control,

assistant professor, Department of Operations Research

Doctoral School of Mathematics

Director: Prof. István Faragó, Doctor of Sciences

Doctoral Program: Applied Mathematics

Director of the Program: Prof. János Karátson, Doctor of Sciences

Department of Operations Research, Faculty of Sciences, Eötvös Loránd University

Budapest, 2018

1 Introduction

We study machine scheduling problems with non-renewable resource constraints. In these problems the jobs have additional resource requirements, and they consume the non-renewable resources when they are started on the machine. The resources have initial stocks, which are replenished at some a-priori known moments of time. We consider the problem on single and parallel machine environments.

Formally, we have a finite set of n jobs, $\mathcal{J} = \{j_1, \ldots, j_n\}$ and a finite set of non-renewable resources $\mathcal{R}$ consumed by the jobs. Each job j has a processing time $p_j \in \mathbb{Z}_+$, and resource requirements $a_{i,j} \in \mathbb{Z}_+$ from the resources $i \in \mathcal{R}$. In case of a single resource we omit the first index and use a_j. Preemption of the jobs is not allowed and each machine can process at most one job at a time. The resources are supplied in q different time moments, $0 = u_1 < u_2 < \cdots < u_q$; the vector $\tilde{b}_\ell \in \mathbb{Z}_+^{|\mathcal{R}|}$ represents the quantities supplied at u_ℓ.

A *schedule* σ specifies a machine and the starting time S_j of each job and it is *feasible* if (i) on every machine the jobs do not overlap in time, and if (ii) at any time point t the total supply from each resource is at least the total request of those jobs starting not later than t, i.e., $\sum_{(\ell \,:\, u_\ell \leq t)} \tilde{b}_{\ell,i} \geq \sum_{(j \,:\, S_j \leq t)} a_{i,j}, \ \forall i \in \mathcal{R}$. We denote the completion time of job j in schedule σ by C_j.

We will consider several well-known objective functions, like the makespan ($C_{\max} := \max_{j\in\mathcal{J}} C_j$), the maximum lateness ($L_{\max} := \max_{j\in\mathcal{J}}(C_j - d_j)$) and the total weighted completion time ($\sum w_j C_j := \sum_{j\in\mathcal{J}} w_j C_j$, where the weight $w_j \in \mathbb{Z}_+$ describes the importance of job j).

Scheduling with non-renewable resources has a great practical interest. We list a few examples from the recent years. Chapter 4 of Stadtler and Kilger (2008) describes examples in the consumer goods industry and in computer assembly, where purchased items have to be taken into account at several planning levels including short-term scheduling which is the topic of the present thesis. Herr and Goel (2016) study a scheduling problem arising in the continuous casting stage of steel production. In Carrera et al. (2010), a similar problem is investigated in a shoe-firm.

2 Terminology

An *optimization problem* Π consists of a set of instances, where each instance has a set of *feasible solutions*, and each solution has an (objective function) value. In a *minimization problem* a feasible solution of minimum value is sought, while in a maximization problem one of maximum value. A $(1 + \varepsilon)$-*approximation algorithm* for a minimization problem Π delivers in polynomial time for each instance of Π a solution whose objective function value is at most $(1+\varepsilon)$ times the optimum value. For a minimization problem Π, a family of approximation algorithms $\{A_\varepsilon\}_{\varepsilon>0}$, where each A_ε is an $(1+\varepsilon)$-approximation algorithm for Π is called a *Polynomial Time Approximation Scheme (PTAS) for* Π. A *Fully Polynomial Time Approximation Scheme* (FPTAS) is a family of algorithms $\{A_\varepsilon\}_{\varepsilon>0}$ with the same properties as a PTAS, plus each A_ε runs in polynomial time in $1/\varepsilon$ as well.

We use the standard $\alpha|\beta|\gamma$ notation for scheduling problems (Graham et al., 1979), where α denotes the processing environment, β the additional restrictions, and γ the objective function. $\alpha = Pm$ indicates m parallel machines for some fixed m. In the β field, 'rm' means that there are non-renewable resource constraints, $rm = r$ indicates $|\mathcal{R}| = r$. Further options are $q = const$ meaning that the number of supplies is a fixed constant and r_j indicates job release dates.

3 Reduction and their consequences

There are strong connections between the variants of our scheduling problem and the variants of the well-known Knapsack Problem (KP). These connections can be described by so-called reductions and they have several important consequences.

In case of one resource there are reductions in both ways between the Knapsack Problem and $1|rm = 1, q = 2|C_{\max}$:

Theorem 3.1 (Györgyi and Kis, 2015a). $1|rm = 1, q = 2|C_{\max} \ \leq_{Strict} KP$.

Theorem 3.2 (Györgyi and Kis, 2015a). $KP \ \leq_{FPTAS} 1|rm = 1, q = 2|C_{\max}$.

The first reduction implies an FPTAS and a fast 3/2 approximation algorithm for $1|rm = 1, q = 2|C_{\max}$:

Corollary 3.3 (Györgyi and Kis, 2015a). *There is an FPTAS for* $1|rm = 1, q = 2|C_{\max}$ *in* $O(n \cdot \min\{\log n, \log(1/\varepsilon)\} + (1/\varepsilon^2)\log(1/\varepsilon) \cdot \min\{n, (1/\varepsilon)\log(1/\varepsilon)\})$ *time and in* $O(n + 1/\varepsilon^2)$ *space.*

Corollary 3.4 (Györgyi and Kis, 2015a). *There is an 3/2-approximation algorithm for* $1|rm = 1, q = 2|C_{\max}$ *of time complexity* $O(n \log n)$.

If we have more than one resource, then there are similar reductions between the r-dimensional Knapsack Problem ($r - DKP$) and $1|rm = r, q = 2|C_{\max}$. From these reductions we can obtain a PTAS for the problem with fixed number of resources and 2 supply dates and we can prove that there is no FPTAS for the problem with at least 2 resources and 2 supply dates unless $P = NP$.

Theorem 3.5 (Györgyi and Kis, 2015a). $1|rm = r, q = 2|C_{\max} \leq_{Strict} r - DKP$.

Corollary 3.6 (Györgyi and Kis, 2015a). *For any fixed r, there is a PTAS for $1|rm = r, q = 2|C_{\max}$.*

Theorem 3.7 (Györgyi and Kis, 2015a). *r-DKP $\leq_{FPTAS}$ $1|rm = r, q = 2|C_{\max}$.*

Corollary 3.8 (Györgyi and Kis, 2015a). *If $r \geq 2$ then there is no FPTAS for $1|rm = r, q = 2|C_{\max}$ unless $P = NP$.*

4 Complexity and approximation results for single machine problems

4.1 Makespan minimization

According to our knowledge the first single machine results of the topic are from the 1980s. The problem was a natural extension of the problem of Carlier and Rinnooy Kan (1982), where there were no machines but there were precedence constraints among the jobs. In case of machine scheduling, the most basic results can be found in Carlier (1984), we highlight the NP-hardness of the problem:

Theorem 4.1 (Carlier, 1984). *The problem $1|rm = 1|C_{\max}$ is NP-hard, even in case of $p_j = a_j$ and $q = 2$.*

Toker et al. (1991) showed a reduction from a variant where the supplies arrive uniformly (i.e., $b_{1,i} = b_{2,i} = \ldots = b_{q,i}$ for each resource i and $u_\ell = (\ell - 1)u_2$) to the two-machine flow shop problem, which is solvable in $O(n \log n)$ time (Johnson, 1954). This result was extended in Xie (1997), which allows several resources. Grigoriev et al. (2005) showed that $1|rm = 1, p_j = p|C_{\max}$ can be solved by scheduling the jobs in non-decreasing resource requirement order as soon as possible. This paper also presented two very simple 2-approximation algorithms for $1|rm|C_{\max}$.

Though the problem $1|rm = 1, q = 2|C_{\max}$ is NP-hard by Theorem 4.1, there is a positive result for a slightly more general problem:

Theorem 4.2 (Györgyi and Kis, 2014). *The problem $1|rm = const, q = const|C_{\max}$ can be solved in pseudo polynomial time.*

The following result of Györgyi and Kis (2017) helps to obtain polynomial time approximation schemes for the general problem $1|rm, r_j|C_{\max}$, provided that we have a family of approximation algorithms for restricted versions of the problem.

Table 1: Approximability of $1|rm|C_{\max}$ with q supplies and $rm = r$ resources if $P \neq NP$. The question mark indicates that we do not know the existence of an FPTAS for $1|rm = 1, q > 2|C_{\max}$.

	$q = 2$	$q > 2$
$r = 1$	FPTAS*	PTAS*, ?
$r = const \geq 2$	PTAS*, no FPTAS*	PTAS, no FPTAS
$r = arb.$	no PTAS	no PTAS

* described in the dissertation

Proposition 4.3 (Györgyi and Kis, 2017). *In order to have a PTAS for* $1|rm, r_j|C_{\max}$*, it suffices to provide a family of algorithms* $\{A_\varepsilon\}_{\varepsilon>0}$ *such that* A_ε *is a* $(1+\varepsilon)$*-approximation algorithm for the restricted problem where the supply dates and the job release dates before* u_q *are from the set* $\{\ell\varepsilon u_q : \ell = 0, 1, 2, \ldots, \lfloor 1/\varepsilon \rfloor\}$.

Now we turn to the achieved approximability results:

Theorem 4.4 (Györgyi and Kis, 2014). *There is a PTAS for* $1|rm = 1, q = const|C_{\max}$.

From Proposition 4.3 and Theorem 4.4, we have the following:

Theorem 4.5. *There is a PTAS for* $1|rm = 1|C_{\max}$.

If $a_j = p_j$ for each job j, then there is a much simpler and faster PTAS for the same problem:

Theorem 4.6 (Györgyi and Kis, 2015b). *There is a PTAS for the problem* $1|rm = 1, a_j = p_j|C_{\max}$.

Györgyi and Kis (2015b) also presented a PTAS for the problem $1|rm = const, q = const|C_{\max}$, which we can generalize by Proposition 4.3 as follows:

Corollary 4.7. *There is a PTAS for the problem* $1|rm = const|C_{\max}$.

We end this section by summarizing the approximability status of the different variants of $1|rm|C_{\max}$ in Table 1.

4.2 Minimizing the total weighted completion time

Minimizing the total completion time is strongly NP-hard. This result was first achieved by Carlier (1984). This paper was written in French, thus it remained unknown for several authors. Later, Gafarov et al. (2011) proved the NP-hardness of the problem and then Kis (2015) rediscovered the result of Carlier:

Table 2: Easy variants of $1|rm = 1|\sum w_j C_j$.

Variant	Optimal schedule
$p_j = a_j = \bar{a}$	non-increasing w_j order
$p_j = w_j = 1$	non-decreasing a_j order
$a_j = w_j = 1$	SPT order
$w_j = \bar{w},\ p_j = a_j$	SPT order
$a_j = \bar{a},\ p_j = w_j$	LPT order

Theorem 4.8 (Carlier, 1984; Kis, 2015)**.** *The problem $1|rm = 1|\sum C_j$ is strongly NP-hard.*

We examine variants where we can state job independent connections among the processing times, the resource requirements and the weights. If these connections are strong enough we can find easy ordering rules that yield optimal schedules, see Table 2.

Then, we provide a non-trivial expression for the objective function value, if $p_j = w_j$ for each job j. In the following lemma H_ℓ denotes the length of the idle period in schedule S in the interval $[u_\ell, u_{\ell+1}]$, and P_ℓ denotes the total working time (when the machine is not idle) in $[u_\ell, u_{\ell+1}]$.

Lemma 4.9 (Györgyi and Kis, 2018b)**.** *If $p_j = w_j$, for each job j, then the objective function value of any feasible schedule S can be expressed as*

$$\sum_j p_j C_j = \sum_{j \le k} p_j p_k + \sum_{\ell=2}^{q} H_{\ell-1} \cdot (P_\ell + P_{\ell+1} + \ldots + P_q).$$

After that, we consider the variant where $p_j = a_j = w_j$ for each job j. Surprisingly, this very restrictive variant is already NP-hard:

Theorem 4.10 (Györgyi and Kis, 2018b)**.** *The problem $1|rm = 1, q = 2, p_j = a_j = w_j|\sum w_j C_j$ is weakly NP-hard, and $1|rm = 1, p_j = a_j = w_j|\sum w_j C_j$ is strongly NP-hard.*

However, we could derive a 2-approximation algorithm for it, using the result of Lemma 4.9:

Theorem 4.11 (Györgyi and Kis, 2018b)**.** *Scheduling the jobs in LPT order is a 2-approximation algorithm for $1|rm = 1, p_j = a_j = w_j|\sum w_j C_j$.*

5 An exact method for $1|rm|L_{\max}$

In contrast to the complexity and approximation results, there are only sporadic computational results on machine scheduling problems with non-renewable resource constraints. Grigoriev et al. (2005) have provided some test results for one of their approximation algorithms. For a related problem, where some of the jobs produce, while other jobs consume some non-renewable resources, Briskorn et al. (2013) propose an exact method for minimizing the total weighted completion time of the jobs. In the more general project scheduling setting, Neumann and Schwindt (2003) study the makespan minimization problem with inventory constraints, and describe a branch-and-bound method for solving it.

This section summarizes the branch-and-cut method of Györgyi and Kis (2018a) for $1|rm|L_{\max}$. Since we have to compute the maximum lateness objective, choosing the right MIP model is a non-trivial issue. After some preliminary tests, we have chosen the model with completion time variables. The MIP formulation is

$$
\begin{array}{llll}
\min L_{\max} & & & \\
\text{s.t.} & & & \\
C_j \geq p_j, & \forall j & \sum_{j \in \mathcal{J}} a_{i,j} z_{j\ell} \leq b_{\ell,i}, & \forall \ell,\ \forall i \\
C_{j_1} + p_{j_2} \leq C_{j_2} + M \cdot (1 - ord_{j_1,j_2}), & \forall j_1 < j_2 & z_{j,\ell-1} \leq z_{j,\ell}, & \forall j,\ \forall \ell \\
C_{j_2} + p_{j_1} \leq C_{j_1} + M \cdot ord_{j_1,j_2}, & \forall j_1 < j_2 & z_{j,q} = 1, & \forall j \\
C_j - p_j \geq \sum_{\ell=2}^{q} u_\ell \cdot (z_{j,\ell} - z_{j,\ell-1}), & \forall j & ord_{j_1,j_2} \in \{0,1\}, & \forall j_1 < j_2 \\
L_{\max} \geq C_j - d_j, & \forall j & z_{j\ell} \in \{0,1\}, & \forall j,\ \forall \ell.
\end{array}
$$

Our branch-and-cut method uses the MIP model as the representation of the problem, and we do not use the solver as a black-box, instead, we generate cutting planes in the course of the solution process in order to speed up the optimization algorithm. We have examined the following cutting planes:

$$
\left.\begin{array}{l}
ord_{j_1,j_2} + ord_{j_2,j_3} - ord_{j_1,j_3} \leq 1, \\
-ord_{j_1,j_2} - ord_{j_2,j_3} + ord_{j_1,j_3} \leq 0
\end{array}\right\} \qquad \forall j_1 < j_2 < j_3,
$$

$$
\sum_{j \in S} p_j (d_{\max}(S) + L_{\max} - C_j + p_j) \geq \frac{1}{2} \left(\sum_{j \in S} p_j^2 + \left(\sum_{j \in S} p_j \right)^2 \right), \qquad S \subseteq \mathcal{J},
$$

$$
L_{\max} \geq C_{j_1} - (p_{j_2} - d_{j_2} + d_{j_1}) \cdot (1 - ord_{j_1,j_2}) + p_{j_2} - d_{j_2}, \qquad \forall j_1 < j_2,
$$

$$
ord_{j_1,j_2} \geq z_{j_1,\ell+1} - z_{j_1,\ell} + z_{j_2,\ell+1} - z_{j_2,\ell} - 1, \qquad \forall j_1, j_2 :\ d_{j_1} < d_{j_2},
$$

$$
\sum_{j \in \mathcal{J}} (z_{j,\ell} - z_{j,\ell-1}) \cdot p_j \leq u_{\ell+1} - u_\ell + p_{max}, \qquad \forall \ell.
$$

Table 3: Results with 50 jobs and 10 supply dates.

50 jobs	1 resource		3 resources		10 resources	
10 supply dates	# opt	avg. gap	# opt	avg. gap	# opt	avg. gap
no cuts	6	1.43	8	1.035	5	1.049
Xpress	7	1.77	7	1.007	5	1.039
our	5	1.20	7	1.021	7	1.029
our + Xpress	7	1.22	6	1.028	7	1.018

We have generated several instances for different (n, q) pairs and for each case, we have examined instances with 1, 3 and 10 resources. The Mosel language of FICO Xpress (2016) was used for implementation. After that, we have compared the results in case of four settings: without generating any cutting planes ('no cuts'), enabling the built-in cuts, but not using our cuts ('Xpress'), disabling the built-in cuts, but using (some of) our cuts ('our'), and using both our and the built-in cuts ('our+Xpress').

Table 3 depicts the results in case of 50 jobs and 10 supply dates for an illustration. We characterize the results by the number of the optimally solved instances (out of 10) and by the average integrality gap (the ratio of the best upper and lower bounds).

6 Results for parallel machine problems

6.1 Makespan minimization

Parallel machine scheduling is also one of the most classical scheduling problems. The makespan minimization problem is already NP-hard in the case of two machines (Lenstra et al., 1977). However, there are several approximation algorithms for this problem, e.g., a 4/3-approximation algorithm by list scheduling (see, e.g., Pinedo (1995)) or a PTAS by Hall and Shmoys (1989). Note that, we can create 7/3-approximation and $(2+\varepsilon)$-approximation algorithms for $P|rm|C_{\max}$ based on these algorithms. This problem was also introduced by Carlier (1984) and in the same year, Slowinski (1984) examined the preemptive version of the problem.

Our first result is a PTAS in case of the number of parallel machines and resources is constant. We have modified this PTAS so that it can deal with the case when (some of) the jobs are dedicated to machines:

Theorem 6.1 (Györgyi and Kis, 2017). *$Pm|rm = const, r_j|C_{\max}$ and $Pm|rm = const, r_j, ddc|C_{\max}$ admit a PTAS.*

Table 4: Approximability of $P|rm|C_{\max}$ with m machines and $rm = r$ resources (in case of $q \geq 2$ supplies) if $P \neq NP$.

	$m = const$	$m = arb.$
$r = 1$	PTAS	PTAS*
$r = const \geq 2$	PTAS*	no PTAS
$r = arb.$	no PTAS	no PTAS

* described in the dissertation

If we have an arbitrary number of parallel machines and at least two resources, then there is no PTAS for the problem (Györgyi and Kis, 2017). However, we have a positive result in the case of one resource:

Theorem 6.2 (Györgyi, 2017)**.** *There is a PTAS for* $P|rm = 1|C_{\max}$.

Table 4 summarizes the approximability status of the different variants of $P|rm|C_{\max}$.

6.2 Lateness minimization

Since the optimum lateness may be 0 or negative, a standard trick is to increase the lateness of the jobs by a constant that depends on the input. In our case, let $L'_{\max} := \max_j\{C_j - d_j + D\}$, where $D := \max_{j\in\mathcal{J}}\{d_j\} + u_q$. In order to provide a PTAS for the lateness objective, we have to assume that the processing times are proportional to the resource consumptions:

Theorem 6.3 (Györgyi and Kis, 2017)**.** *If* $L'_{\max}$ *is defined as above, then* $Pm|rm = 1, p_j = a_j|L'_{\max}$ *admits a PTAS.*

Acknowledgment

Between 2013 and 2016, I was a state supported PhD student at ELTE. Since 2016, I work at MTA SZTAKI and my research was supported by the GINOP-2.3.2-15-2016-00002 grant and by the Young Researcher Scholarship of MTA.

The thesis based on the following papers:

Györgyi, P. (2017). A PTAS for a resource scheduling problem with arbitrary number of parallel machines. *Operations Research Letters*, 45(6):604 – 609.

Györgyi, P. and Kis, T. (2014). Approximation schemes for single machine scheduling with non-renewable resource constraints. *Journal of Scheduling*, 17:135–144.

Györgyi, P. and Kis, T. (2015a). Reductions between scheduling problems with non-renewable resources and knapsack problems. *Theoretical Computer Science*, 565:63–76.

Györgyi, P. and Kis, T. (2015b). Approximability of scheduling problems with resource consuming jobs. *Annals of Operations Research*, 235(1):319–336.

Györgyi, P. and Kis, T. (2017). Approximation schemes for parallel machine scheduling with non-renewable resources. *European Journal of Operational Research*, 258(1):113–123.

Györgyi, P. and Kis, T. (2018a). Minimizing the maximum lateness on a single machine with raw material constraints by branch-and-cut. *Computers & Industrial Engineering*, 115:220 – 225.

Györgyi, P. and Kis, T. (2018b). Minimizing total weighted completion time on a single machine subject to non-renewable resource constraints. *Submitted*.

References

Briskorn, D., Jaehn, F., and Pesch, E. (2013). Exact algorithms for inventory constrained scheduling on a single machine. *Journal of Scheduling*, 16(1):105–115.

Carlier, J. (1984). *Problèmes d'ordonnancement à contraintes de ressources: algorithmes et complexité*. Université Paris VI-Pierre et Marie Curie, Institut de programmation.

Carlier, J. and Rinnooy Kan, A. H. G. (1982). Scheduling subject to nonrenewable-resource constraints. *Operations Research Letters*, 1(2):52–55.

Carrera, S., Ramdane-Cherif, W., and Portmann, M.-C. (2010). Scheduling supply chain node with fixed component arrivals and two partially flexible deliveries. In *5th International Conference on Management and Control of Production and Logistics-MCPL 2010*, page 6. IFAC Publisher.

FICO Xpress (2016). Optimization suite. www.fico.com/xpress.

Gafarov, E. R., Lazarev, A. A., and Werner, F. (2011). Single machine scheduling problems with financial resource constraints: Some complexity results and properties. *Mathematical Social Sciences*, 62:7–13.

Graham, R. L., Lawler, E. L., Lenstra, J. K., and Rinnooy Kan, A. H. G. (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, 5:287–326.

Grigoriev, A., Holthuijsen, M., and van de Klundert, J. (2005). Basic scheduling problems with raw material constraints. *Naval Research of Logistics*, 52:527–553.

Hall, L. A. and Shmoys, D. B. (1989). Approximation schemes for constrained scheduling problems. In *Proceedings of the 30th Annual Symposium on Foundations of Computer Science*, pages 134–139. IEEE.

Herr, O. and Goel, A. (2016). Minimising total tardiness for a single machine scheduling problem with family setups and resource constraints. *European Journal of Operational Research*, 248:123–135.

Johnson, S. M. (1954). Optimal two-and three-stage production schedules with setup times included. *Naval Research Logistics (NRL)*, 1(1):61–68.

Kis, T. (2015). Approximability of total weighted completion time with resource consuming jobs. *Operations Research Letters*, 43(6):595–598.

Lenstra, J. K., Rinnooy Kan, A. H. G., and Brucker, P. (1977). Complexity of machine scheduling problems. *Annals of Discrete Mathematics*, 1:343–362.

Neumann, K. and Schwindt, C. (2003). Project scheduling with inventory constraints. *Mathematical Methods of Operations Research*, 56(3):513–533.

Pinedo, M. (1995). *Scheduling – Theory, Algorithms, and Systems*. Prentice Hall, Englewood Cliffs, New Jersey.

Slowinski, R. (1984). Preemptive scheduling of independent jobs on parallel machines subject to financial constraints. *European Journal of Operational Research*, 15:366–373.

Stadtler, H. and Kilger, C. (2008). *Supply Chain Management and Advanced Planning. Concepts, Models, Software, and Case Studies*. Springer, 4th edition.

Toker, A., Kondakci, S., and Erkip, N. (1991). Scheduling under a non-renewable resource constraint. *Journal of the Operational Research Society*, 42:811–814.

Xie, J. (1997). Polynomial algorithms for single machine scheduling problems with financial constraints. *Operations Research Letters*, 21(1):39–42.