Bilevel scheduling on a single machine in an adversarial setting

T'kindt V.[1], Della Croce F.[2] and Agnetis A.[3]

[1] Université de Tours, Laboratoire d'Informatique Fondamentale et Appliquée (EA 6300), ERL CNRS 7002 ROOT, Tours, France,
tkindt@univ-tours.fr

[2] DIGEP, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, CNR, IEIIT, Torino, Italy.
federico.dellacroce_a@polito.it

[3] Università degli Studi di Siena, Dipartimento di Ingegneria dell'Informazione e Scienze Matematiche, via Roma 56, 53100 Siena, Italy.
agnetis@diism.unisi.it

Mots-clés : *Scheduling, bilevel optimization.*

1 Introduction

In this contribution we focus on a particular setting in which two agents are concerned by the scheduling of a set of n jobs. The first agent, called the *leader*, can take some decisions before providing the jobset to the second agent, called the *follower*, who then takes the remaining decisions to solve the problem. As an example, the leader could select a subset of $n' \leq n$ jobs that the follower has to schedule. Notice that the decisions the agents can take are exclusive : in this example, the follower cannot decide the jobs to schedule and the leader cannot schedule the jobs. This setting falls into the category of *bilevel optimization* [4]. In such problems it is assumed that the leader and the follower follow their own objectives which can be contradictory, so leading to very hard optimization problems. Recently, many papers on bilevel combinatorial optimization appeared, here we refer to [2, 3, 6, 5, 7, 10] just to mention a few. On the other hand, to the authors knowledge, the literature on bilevel scheduling is much more limited. We refer here to [1, 8, 9]. We focus in the following on single machine scheduling under the adversarial framework where the goal of the leader is to make the follower solution as bad as possible and provide several exact polynomial time algorithms for different objective functions.

2 Adversarial bilevel single machine scheduling

It is assumed that n jobs are to be scheduled on a single disjunctive machine. Each job j is defined by a processing time p_j and, depending on the problem, a weight w_j or a due date d_j. The follower is scheduling jobs so that its objective function $f^F \in \{\sum_j C_j^F, \sum_j w_j^F C_j^F, L_{max}^F, \sum_j U_j^F\}$ is minimized. Beforehand, the leader can some decisions which impact the instance solved by the follower. We consider two different scenario :

(S1) The leader selects a subset of $n' \leq n$ jobs, for any given n', that the follower schedules.

(S2) The leader decides of quantities q_j so that the processing times or the weights or the due dates are modified. The leader has a given budget $Q \in \mathbb{N}$ so that $\sum_j |q_j| \leq Q$. So, if the processing times p_j are modified by the leader, the follower schedules jobs with processing times $p_j^F = p_j + q_j$.

The leader takes decisions so that the optimal solution computed by the follower is as bad as possible. Considering the three-field notation for scheduling problems, we will denote by $ADV-n$ the problems in which the leader selects a subset of jobs (scenario (S1)) and $ADV-p$

the problems in which the leader modifies only the processing times (scenario (S2)). Similarly, $ADV-w$ (resp. $ADV-d$) refers to the problems in which only the weights (resp. the due dates) are modified. The proposed results, discussed during the conference, are summarized in Table 1.

Polynomially solvable problems	
$1\|ADV-n\|\sum_j C_j^F$	$1\|ADV-n\|L_{max}^F$
$1\|ADV-n\|\sum_j U_j^F$	$1\|ADV-p\|\sum_j C_j^F$
$1\|ADV-p,q_j\in\mathbb{R}\|\sum_j w_j^F C_j^F$	$1\|ADV-w,q_j\in\mathbb{R}\|\sum_j w_j^F C_j^F$
$1\|ADV-p\|L_{max}^F$	$1\|ADV-p,d_j=d\|\sum_j U_j^F$
$1\|ADV-d,d_j=d\|\sum_j U_j^F$	
$\mathcal{NP}$-hard problem	
$1\|ADV-n,\mathcal{L}\|-$	
Open problems	
$1\|ADV-n\|\sum_j w_j C_j^F$	$1\|ADV-p,q_j\in\mathbb{N}\|\sum_j w_j^F C_j^F$
$1\|ADV-w,q_j\in\mathbb{N}\|\sum_j w_j^F C_j^F$	$1\|ADV-p\|\sum_j U_j^F$
$1\|ADV-d\|\sum_j U_j^F$	$1\|ADV-d\|L_{max}^F$

TAB. 1 – Complexity status of some bilevel single machine scheduling problems

Acknowledgements

This work has been partially supported by "Ministero dell'Istruzione, dell'Università e della Ricerca" Award "TESUN-83486178370409 finanziamento dipartimenti di eccellenza CAP. 1694 TIT. 232 ART. 6".

References

[1] Abass S.A. : Bilevel programming approach applied to the flow shop scheduling problem under fuzziness. Computational Management Science. 2 : 279–293 (2005)

[2] Caprara, A., Carvalho, M., Lodi, A., Woeginger, G. : Bilevel Knapsack with Interdiction Constraints. INFORMS Journal on Computing. 28, 319–333 (2016)

[3] Della Croce, F., Scatamacchia, R. : An exact approach for the bilevel knapsack problem with interdiction constraints and extensions. Mathematical Programming. 183, 249–281 (2020).

[4] Dempe, S., Kalashnikov, V. Perez-Valdes, G.A., Kalashnikova, N., 2015, "Bilevel programming problems", *Springer*.

[5] Fischetti, M., Ljubić, I., Monaci, M., Sinnl, M. : Interdiction Games and Monotonicity, with Application to Knapsack Problems. INFORMS Journal on Computing. 31, 390–410 (2019)

[6] Fischetti, M., Ljubić, I., Monaci, M., Sinnl, M. : A New General-Purpose Algorithm for Mixed-Integer Bilevel Linear Programs. Operations Research. 65, 1615–1637 (2017)

[7] Fischetti, M., Ljubić, I., Monaci, M., Sinnl, M. : On the use of intersection cuts for bilevel optimization. Mathematical Programming. 172, 77–103 (2018)

[8] Karlof, J.K., Wang, W. : Bilevel programming applied to the flow shop scheduling problem. Computers and Operations Research. 23 :5, 443–451 (1996).

[9] Kis, T., Kovacs, A. : On bilevel machine scheduling problems. OR Spectrum. 34, 43–68 (2012)

[10] Woeginger, G. : The trouble with the second quantifier. 4OR. 19, 157–181 (2021)