Open shop problem with agreement graph: new results

Nour ElHouda TELLACHE[1], Mourad BOUDHAR[2] and Farouk YALAOUI[3]

[1] CERMICS Laboratory, ENPC, 6-8 avenue Blaise Pascal, 77455 Marne-la-Vallée Cedex 2, France
nour.tellache@gmail.com
[2] RECITS Laboratory, USTHB university, BP 32 El-Alia,Bab-Ezzouar, Algiers, Algeria
mboudhar@yahoo.fr
[3] LOSI Laboratory, UTT university, 12 rue Marie Curie BP 2060, 10010, Troyes Cedex, France
farouk.yalaoui@utt.fr

Abstract. This paper deals with the problem of scheduling on a two-machine open shop subject to constraints given by an agreement graph G, such that jobs can be processed simultaneously on different machines if and only if they are represented by adjacent vertices in G. The problem of minimizing the maximum completion time (makespan) is known to be NP-hard. In this work, we study the complexity of the problem when restricted to trees. Then, we present six Mixed Integer Linear Programming (MILP) models along with an experimental study to test their performance.

Keywords: open shop scheduling, agreement graph, complexity, MILP, makespan.

1 Introduction

The Open Shop problem with Agreement graph (OSA) which is discussed in this paper can be described as follows. The inputs consist of a finite set $\{J_j, j = 1, \ldots, n\}$ of n jobs that has to be processed on a set $\{M_i, i = 1, \ldots, m\}$ of m machines and a simple graph $G = (V, E)$ over the jobs, called the agreement graph. Each job J_j consists of m operations J_{ij} $(i = 1, \ldots, m)$, where J_{ij} has to be processed on machine M_i for $p_{ij} \geq 0$ time units. The order in which the jobs are processed on the machines is not fixed. On the other hand, each vertex in G represents a job and two jobs can be processed at the same time on different machines (are not in conflict) if and only if they are adjacent in G. The objective is to find a feasible schedule that minimizes the maximum completion time (makespan). According to the three field classification $\alpha/\beta/\gamma$ of Graham et al. [4], we denote our scheduling problem by $O2|AgreeG = (V, E)|C_{max}$, where $AgreeG = (V, E)$ indicates the presence of an agreement graph $G = (V, E)$ over the jobs. The proportionate processing times assumption implies that each job J_j has the same processing requirement p_j on each machine ($p_{ij} = p_j$ for all jobs J_j and all machines M_i); in that case, the resulting problem is called the proportionate OSA problem and it is denoted $Om|AgreeG = (V, E), prpt|C_{max}$.

In some practical applications, the jobs may require, besides the machines, some additional non-sharable resources with limited capacities for their processing [1]. In this case, two jobs can be processed simultaneously on different machines if the total requirement of at least one resource does not exceed its capacity. Therefore, this problem can be modeled as an OSA problem. For more details about the correspondence between the OSA problem and the open shop under resource constraints, the interested reader is referred to [6].

2 Literature review

Scheduling with agreement graph $G = (V, E)$ is equivalent to scheduling with conflict graph $\overline{G} = (V, \overline{E})$ (complement of the agreement graph). Tellache and Boudhar [6] studied the Open Shop problem with Conflict graph (OSC). They showed that the two-machine OSC problem with $p_{ij} \in \{1, 2, 3\}$ is NP-hard in the strong sense even when restricted to complements of bipartite graphs. The same result holds for the three-machine OSC problem with $p_{ij} = 1$ and an arbitrary conflict graph. After that, efficient algorithms were proposed for the two-machine OSC problem with $p_{ij} \in \{0, 1, 2\}$, and for the three-machine OSC problem with $p_{ij} = 1$ and $\overline{G}$ being a complement of triangle-free graph. They also proved that by allowing preemption, the two-machine OSC problem becomes easy to solve for arbitrary conflict graphs. On the other hand, they found that the OSC problem is polynomially equivalent to a special case of the open shop under resource constraints, from which new complexity results of the latter problem were established. They also presented a two-phase heuristic approach and lower bounds for the general m-machine OSC problem. In [5], the authors considered the same problem. They first proved the NP-hardness of the case of two values of processing times and more general agreement graphs which closes definitely the complexity status of the problem. Then, they presented some restricted cases that can be solved in polynomial time. They also derived new complexity results of the open shop under resource constraints and of the partition into triangles problem.

3 NP-hardness results

In this Section, we show that the OSA problem is NP-hard even when restricted to trees. The problem used in the reduction process is the 2-partition problem [3].

Theorem 1. *The problem $O2|AgreeG = (V, E)|C_{max}$ is NP-hard in the ordinary sense for G being a tree.*

In the following theorem, we consider the proportionate open shop problem.

Theorem 2. *The problem $O2|AgreeG = (V, E), prpt|C_{max}$ is NP-hard in the ordinary sense for G being a tree.*

4 Mathematical models

The following models are proposed for the problem $O|AgreeG = (V, E)|C_{max}$. The parameters used are:

- a_{jk}: 1 if J_j and J_k are in conflict, 0 otherwise.
- M: big constant.

The decision variables of the first model are:

- C_{max}: the maximum completion time determined by the completion time of the last operation.
- C_{ij}: completion time of job J_j on machine M_i.
- x_{ijk}: 1 if job J_j is scheduled any time before J_k on machine M_i, 0 otherwise.
- $y_{ii'j}$: 1 if job J_j is scheduled on M_i then on $M_{i'}$, 0 otherwise.
- $r^{jk}_{ii'}$: when $a_{jk} = 1$, this variable is equal to 1 if the operation J_{ij} is scheduled any time before $J_{i'k}$, 0 otherwise..

The MILP model is summarized in (P_1).

$$
(P_1)\left|\begin{array}{llll}
\text{min } C_{max} & & & \\
\text{S.C } C_{max} \geq C_{ij}; & i=1,\ldots,m; & j=1,\ldots,n, & (1)\\
\quad C_{ij} - M(1-x_{ikj}) \leq C_{ik} - p_{ik}; & i=1,\ldots,m, & 1 \leq j < k \leq n, & (2)\\
\quad C_{ij} - C_{ik} - p_{ij} \geq -Mx_{ikj}; & i=1,\ldots,m, & 1 \leq j < k \leq n, & (3)\\
\quad C_{ij} - M(1-y_{ii'j}) \leq C_{i'j} - p_{i'j}; & 1 \leq i' < i \leq m, & j=1,\ldots,n, & (4)\\
\quad C_{ij} - C_{i'j} - p_{ij} \geq -My_{ii'j}; & 1 \leq i' < i \leq m, & j=1,\ldots,n, & (5)\\
\quad C_{ij} - M(1-r_{iji'k}) \leq C_{i'k} - p_{i'k}; & \text{if } a_{jk}=1,\ (i \neq i'), i,i'=1,\ldots,m,, & j=1,\ldots,n, & (6)\\
\quad C_{ij} - C_{i'k} - p_{ij} \geq -Mr_{iji'k}; & \text{if } a_{jk}=1,\ (i \neq i'), i,i'=1,\ldots,m,, & j=1,\ldots,n, & (7)\\
\quad x_{ijk}, y_{ii'j} \in \{0,1\}; & 1 \leq i' < i \leq m, & 1 \leq j < k \leq n, & (8)\\
\quad r_{iji'k} \in \{0,1\}; & (i \neq i'), i,i'=1,\ldots,m, & 1 \leq j < k \leq n, & (9)\\
\quad C_{ij} \geq p_{ij}; & i=1,\ldots,m, & j=1,\ldots,n. & (10)
\end{array}\right.
$$

(1) equates the makespan to the maximum of the completion times of all operations.

(2) and (3) ensure that job J_k either precedes job J_j or follows it on M_i, but not both.

(4) and (5) ensure that the operations of the same job cannot be processed at the same time on different machines.

(6) and (7) ensure that two conflicting jobs cannot be processed simultaneously on different machines.

The conflict constraints between the jobs and the conflicts between the operations of the same job can be modelled without introducing the variables $y_{ii'j}$ and $r_{iji'k}$ as follows.

$$
(P_2)\left|\begin{array}{llll}
\text{min } C_{max} & & & \\
\text{S.C } C_{max} \geq C_{ij}; & i=1,\ldots,m; & j=1,\ldots,n, & (1)\\
\quad C_{ij} - M(1-x_{ikj}) \leq C_{ik} - p_{ik}; & i=1,\ldots,m, & 1 \leq j < k \leq n, & (2)\\
\quad C_{ij} - C_{ik} - p_{ij} \geq -Mx_{ikj}; & i=1,\ldots,m, & 1 \leq j < k \leq n, & (3)\\
\quad \frac{\max\{c_{ij}-c_{i'j};0\}}{p_{ij}} + \frac{\max\{c_{i'j}-c_{ij};0\}}{p_{i'j}} \geq 1; & 1 \leq i' < i \leq m, & j=1,\ldots,n, & (11)\\
\quad \frac{\max\{c_{ij}-c_{i'k};0\}}{p_{ij}} + \frac{\max\{c_{i'k}-c_{ij};0\}}{p_{i'k}} \geq 1; & \text{if } a_{jk}=1,\ (i \neq i'), i,i'=1,\ldots,m, & 1 \leq j < k \leq n, & (12)\\
\quad x_{ijk} \in \{0,1\}; & i=1,\ldots,m & 1 \leq j < k \leq n, & (13)\\
\quad C_{ij} \geq p_{ij}; & i=1,\ldots,m, & j=1,\ldots,n. & (10)
\end{array}\right.
$$

We replaced constraints (4) and (5) of (P_1) by constraint (11) of (P_2) and (6) and (7) of (P_1) by the constraint (12) of (P_2).

The disjoint constraints of (P_1) and (P_2) can be written in two different ways:

– Combine each pair of inequality dichotomous constraints into a single equality constraint that we set equal to a surplus variable as follows:

$$
(2)+(3) \Rightarrow \begin{cases} C_{ij} - C_{ik} + Mx_{ikj} - p_{ij} = X_{ijk}; & i=1,\ldots,m, 1 \leq j < k \leq n\\ M - p_{ik} - p_{ij} \geq X_{ijk}; & i=1,\ldots,m, 1 \leq j < k \leq n\\ X_{ijk} \geq 0; & i=1,\ldots,m, 1 \leq j < k \leq n \end{cases}
$$

$$
(4)+(5) \Rightarrow \begin{cases} C_{ij} - C_{i'j} + My_{ii'j} - p_{ij} = Y_{ii'k}; & 1 \leq i' < i \leq m, j=1,\ldots,n\\ M - p_{i'j} - p_{ij} \geq Y_{ii'k}; & 1 \leq i' < i \leq m, j=1,\ldots,n\\ Y_{ii'k} \geq 0; & 1 \leq i' < i \leq m, j=1,\ldots,n \end{cases}
$$

$$
(6)+(7) \Rightarrow \begin{cases} C_{ij} - C_{i'k} + Mr_{iji'k} - p_{ij} = R_{iji'k}; & \text{if } a_{jk}=1, (i \neq i'), i,i'=1,\ldots,m, 1 \leq j < k \leq n\\ M - p_{i'k} - p_{ij} \geq R_{iji'k}; & \text{if } a_{jk}=1, (i \neq i'), i,i'=1,\ldots,m, 1 \leq j < k \leq n\\ R_{iji'k} \geq 0; & \text{if } a_{jk}=1, (i \neq i'), i,i'=1,\ldots,m, 1 \leq j < k \leq n \end{cases}
$$

By replacing these constraints in (P_1) and (P_2), we obtain the models (P_3) and (P_4) respectively.

– Keep the first inequality and add the fact that the sum of the two variables equals 1.

$$(2)+(3) \Rightarrow \begin{cases} C_{ij} - M(1-x_{ikj}) \leq C_{ik} - p_{ik}; & i=1,\ldots,m, 1 \leq j \neq k \leq n \\ x_{ijk} + x_{ikj} = 1; & i=1,\ldots,m, 1 \leq j \neq k \leq n \end{cases}$$

$$(4)+(5) \Rightarrow \begin{cases} C_{ij} - M(1-y_{ii'j}) \leq C_{i'j} - p_{i'j}; & 1 \leq i' \neq i \leq m, j=1,\ldots,n \\ y_{ii'j} + y_{i'ij} = 1; & 1 \leq i' \neq i \leq m, j=1,\ldots,n \end{cases}$$

$$(6)+(7) \Rightarrow \begin{cases} C_{ij} - M(1-r_{iji'k}) \leq C_{i'k} - p_{i'k}; & 1 \leq i' \neq i \leq m, 1 \leq j \neq k \leq n \\ r_{iji'k} + r_{i'kij} = 1; & 1 \leq i' \neq i \leq m, 1 \leq j \neq k \leq n \end{cases}$$

$$(8)+(9) \Rightarrow \left\{ x_{ijk}, y_{ii'j}, r_{iji'k} \in \{0,1\}; 1 \leq i' \neq i \leq m, 1 \leq j \neq k \leq n \right.$$

$$(13) \Rightarrow \left\{ x_{ijk} \in \{0,1\}; 1 \leq i' \neq i \leq m, 1 \leq j \neq k \leq n \right.$$

By replacing these constraints in (P_1) and (P_2), we obtain the models (P_5) and (P_6) respectively.

5 Computational experiments

The runs of the above mathematical models were made with Microsoft Visual Studio 2017 (using C++ language) and the models were solved using Cplex 12.8 solver. All experiments were carried out on randomly generated instances. The conflict graph of each instance is generated using the $G(n,p)$ Erdős Rényi method [2], where p is the probability that an edge exists between two vertices. The number of jobs we considered is from 3 to 11 and the number of machines $m \in \{2,3,5\}$. We also considered three values of p, $p \in \{0.2, 0.5, 0.8\}$. The processing times of the jobs were randomly generated from a uniform distribution in the interval $[0,100]$. Note that M in the experiments is set $M = \sum_{i=1}^{m}\sum_{j=i}^{n} p_{ij}$.

We observed from the implementation that the MILP models based on (P_1) ((P_1), (P_3) and (P_5)) require less CPU time than the models based on (P_2) ((P_2), (P_4) and (P_6)). Regarding the modeling of the disjoint constraints, we observed that the first and third types of constraints perform better than combining the inequalities into an equality constraint that we set equal to a surplus variable.

6 Perspectives

For future perspectives, more research is needed concerning the complexity of the OSA problem when restricted to other particular graphs. Also, more research and numerical simulations are needed to enhance the mathematical models, e.g. by adding valid cuts and by introducing procedures to improve the value of big M.

Bibliography

[1] J. Blazewicz, W. Cellary, R. Slowinski, and J. Weglarz. Scheduling under resource constraints-deterministic models. *Annals of Operations Research*, 7, 1986.

[2] P. Erdős and A. Rényi. On random graphs 1. *Publicationes Mathematicae*, 6:290–297, 1959.

[3] M. R. Garey and D. S. Johnson. *Computers and intractability: A Guide to the Theory of NP-Completeness*. New York: Freeman, 1979.

[4] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of discrete mathematics*, 5:287–326, 1979.

[5] N. Tellache, M. Boudhar, and F. Yalaoui. Two-machine open shop problem with agreement graph. *Theoretical Computer Science*, 796:154–168, 2019.

[6] N. E. H. Tellache and M. Boudhar. Open shop scheduling problems with conflict graphs. *Discrete Applied Mathematics*, 227:103 – 120, 2017.