Computers and Operations Research 98 (2018) 231–239

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

Distributionally robust fixed interval scheduling on parallel identical machines under uncertain finishing times

Martin Branda

Faculty of Mathematics and Physics, Department of Probability and Mathematical Statistics, Charles University, Sokolovská 83, Prague 186 75, Czech Republic

ARTICLE INFO

Article history:
Received 9 February 2017
Revised 29 April 2018
Accepted 29 May 2018
Available online 5 June 2018

MSC:
90C15
90B36
90C10

Keywords:
Fixed interval scheduling
Stochastic job durations
Distributional robustness
Cardinality constrained ambiguity set
Decomposition algorithm

ABSTRACT

We deal with fixed interval scheduling (FIS) problems on parallel identical machines where the job starting times are given but the finishing times are subject to uncertainty. In the operational problem, we construct a schedule with the highest worst-case probability that it remains feasible, whereas in the tactical problem we are looking for the minimum number of machines to process all jobs given a minimum level for the worst-case probability that the schedule is feasible. Our ambiguity set contains joint delay distributions with a given copula dependence, where a proportion of marginal distributions is stressed and the rest are left unchanged. We derive a trackable reformulation and propose an efficient decomposition algorithm for the operational problem. The algorithm for the tactical FIS is based on solving a sequence of the operational problems. The algorithms are compared on simulated FIS instances in the numerical part.

1. Introduction

In this paper, we deal with distributionally robust fixed interval scheduling problems that can be characterized as follows. The number of jobs together with their start and prescribed end times, when each job is expected to be finished, are known in advance. However, the actual end times are subject to uncertainty which is taken into account using random delays. Preemption of jobs is not allowed; once a job is assigned to a machine, it must be finished by that machine and cannot be transferred to another machine. Moreover, no job is allowed to be preempted and resumed later on the same machine. Each job must be processed by one machine, and each machine can process at most one job at a time. All machines are identical, i.e., the processing times including the distribution of random delays do not depend on the processing machine. The probabilistic distribution is not known precisely, and it is assumed to belong to an ambiguity set which contains possible distributions of random delays. The problems lie in finding "here-and-now" decisions on assigning the jobs to machines at the beginning of the planning horizon. If the schedule becomes infeasible during its processing, a re-optimization is necessary. Note that an earlier end, which is also possible in practice, does not influence the schedule feasibility because the starting times of the subsequent jobs are given and cannot be moved. Thus, only positive delays can cause problems with the schedule feasibility.

We will focus on two variants of the robust FIS problem. In the operational problem, we would like to construct a schedule with the highest worst-case probability that it remains feasible. In the tactical problem, we are looking for the minimum number of machines to process all jobs given a minimum level for the worst-case probability that the schedule is feasible. The worst-case probability is related to the most pessimistic choice of the probability distribution from a particular ambiguity set. The stochastic integer programming formulations of these problems are provided in Section 2.

We will discuss a possible application to the gate assignment problem where a set of flights is assigned to available gates. Based on historical observations, we can estimate the parameters of the delay distribution. However, there can be a proportion of gate assignments, possibly time-period dependent, for which a worse delay appears. This can be incorporated into the FIS problem by a suitable choice of the ambiguity set of the delay distributions. The operational problem then leads to a flight assignment to available gates with the highest attainable worst-case probability that the schedule remains feasible, whereas the tactical problem aims at

E-mail address: branda@karlin.mff.cuni.cz

https://doi.org/10.1016/j.cor.2018.05.025
0305-0548/

finding the optimum number of gates to fulfill a prescribed worst-case probability for the schedule feasibility.

We briefly review the recent literature on FIS. Older results on deterministic FIS problems were summarized by two survey papers Kolen et al. (2007), Kovalyov et al. (2007). Eliiyi (2013) discussed complexity and algorithms for the operational and tactical problems and identified several polynomially solvable special cases. Angelelli et al. (2014) compared exact and heuristic approaches to solving a FIS problem with additional resource constraints, which was earlier shown to be NP-hard by Angelelli and Filippi (2011). Ng et al. (2014) reduced the FIS problem to finding a maximum weight clique in a graph and proposed an exact algorithm and several greedy heuristics. Zhou et al. (2014) dealt with the tactical problem with machine classes and spread-time constraints and developed a specialized branch-and-price algorithm. The tactical FIS problem is also related to the shift minimization personnel task scheduling problem, see Lin and Ying (2014), Smet et al. (2014).

There are three recent papers dealing with FIS problems under uncertain processing times using stochastic integer programming formulations. Branda et al. (2015) proposed a two-stage stochastic programming formulation which aims at maximizing the reward for processing the selected jobs and at the same time minimizing the costs for outsourcing additional machines. An operational problem devoted to optimal scheduling of all jobs to available machines was considered by Branda et al. (2016), who proposed an extended robust coloring reformulation of the FIS problem with random delays and suggested a tabu search algorithm for solving larger instances. Branda and Hájek (2017) observed that a network-based formulation enables us to optimally solve a larger instance by the mixed-integer programming solver IBM Cplex. Moreover, this formulation enabled the consideration of heterogeneous machines and job delays dependent on a selected machine.

The basic stochastic programming approach assumes the full knowledge of the probability distribution of random parts. However, in many real-life stochastic optimization problems, the probability distribution is not known precisely and the decision-maker must rely on its approximation. Therefore, it is necessary to investigate stability and robustness of the obtained solutions with respect to the choice of the distribution. Since the seminal work of Žáčková (1966), the minimax approach to robustness has obtained high attention in the stochastic programming literature. In this approach, the decision-maker hedges against the worst-case (the most pessimistic) distribution taken from a set of probability distributions called the ambiguity set. Previous works on distributionally robust scheduling employed moment conditions where the first two moments are given, see, e.g., Chang et al. (2017), Wiesemann et al. (2012). This paper proposes the first attempt to deal with distributionally robust FIS problems, moreover under a cardinality constrained ambiguity set. This set contains all distributions of delays where the joint distribution follows an Archimedean copula whereas a fraction of the marginals is stressed and the rest are left unchanged.

We deal with robust operational and tactical problems under ambiguous sets of probability distributions. We combine several results valid for the robust combinatorial problems and the network flow problems from Bertsimas and Sim (2003), who considered the cardinality constrained uncertainty set to deal with robustness with respect to the uncertain coefficients in the objective function. This uncertainty set was also considered by Bertsimas and Sim (2004), who gave a probabilistic interpretation for this choice as well. Since we deal with probabilistic distributions, we will use the more appropriate term of a cardinality constrained ambiguity set. In our case, the cardinality constrained ambiguity set enables us to look into the robustness and stability of the optimal values and solutions with respect to perturbations of the marginal distributions.

The maximization of the worst-case probability in the operational FIS leads to a minimax problem which is directly intractable. However, under our assumptions, we are able to find its tractable reformulation which leads to a cost–Conditional Value at Risk (CVaR) minimization under network flow constraints. This reformulation can be solved directly by a mixed-integer programming solver. Since this approach is already demanding, we propose a decomposition algorithm. We will obtain a second-stage (slave) problem which can be solved as a network flow problem with relaxed binary variables. The master problem can then be solved by the evaluation algorithm introduced by Bertsimas and Sim (2003) or, as we propose, by the golden-section search method, cf. Bazaraa et al. (2006). These approaches to the solution are then compared in the empirical study showing a significant improvement in the computational time using the new method.

The robust tactical problem belongs to a general class of chance constrained problems under ambiguity. In this area, mainly the moment conditions were considered to define the ambiguity set, see, e.g., Hanasusanto et al. (2015), Zymler et al. (2013). We will benefit from a special structure of the tactical FIS problem under the cardinality constrained ambiguity set. We propose a modified binary search over the number of machines where we solve an operational problem in each of the iterations. Moreover, we will investigate the influence of the copula dependence on the optimal number of machines in the numerical part.

This paper is organized as follows. In Section 2, we provide mathematical formulations of the distributionally robust FIS problems and formulate the distributional assumptions. In Section 3, we propose the reformulation of the robust operational problem and discuss the algorithms based on its decomposition. An algorithm for solving the robust tactical problem is then introduced in Section 4. In Section 5, we provide an extensive numerical study based on simulated instances. Section 6 concludes the paper.

2. Stochastic programming formulations

In this section, we propose formulation of the distributionally robust FIS problems and discuss the distributional assumptions. We will consider m parallel identical machines and n jobs. For each job j there is a known starting time s_j. We denote by $\mathcal{S} = \{s_1, \dots, s_n\}$ the set of all starting times when it is necessary to verify that at most one job is assigned to each machine, cf. Kroon et al. (1995, 1997). We assume that the finishing time is uncertain and it can be written as $f_j(\xi) = f_j + D_j(\xi)$, where f_j is a prescribed finishing time of the job j and $D_j(\xi)$ denotes a random delay which is a nonnegative random variable on probability space $(\Xi, \mathcal{A}, P)$, where Ξ is the set of elementary events, $\mathcal{A}$ denotes a $\sigma-$algebra and P is a probability measure. However, we assume that the probability measure P is not known precisely but it belongs to an ambiguity set $\mathcal{P}$.

The robust operational FIS can be formulated as a minimax problem where the worst-case probability that the schedule remains feasible, i.e., there is at most one job assigned to each machine at all starting times, is maximized:

$$\max_{x_{ji}} \min_{P \in \mathcal{P}} P\left(\xi \in \Xi : \sum_{\{j:\, s_j \le t < f_j(\xi)\}} x_{ji} \le 1,\ t \in \mathcal{S},\ i = 1, \dots, m \right) \tag{1}$$

$$\sum_{\{j:\, s_j \le t < f_j\}} x_{ji} \le 1, \quad i = 1, \dots, m, \quad t \in \mathcal{S}, \tag{2}$$

$$\sum_{i=1}^{m} x_{ji} = 1, \quad j = 1, \dots, n, \tag{3}$$

$$x_{ji} \in \{0, 1\}, \quad i = 1, \dots, m, \quad j = 1, \dots, n, \tag{4}$$

where $\sum_{\{j:\, s_j \le t < f_j\}}$ represents the sum over all jobs for which the interval $[s_j, f_j)$ contains t from the set $\mathcal{S}$. If $s_k \ge f_j(\xi)$, then jobs j and k can be processed by the same machine. Binary decision variables x_{ji} are used to assign jobs j to machines i. The first constraints ensure that at most one job is assigned to a machine at each time $t \in \mathcal{S}$ with respect to the prescribed processing intervals. The random delays are taken into account in the objective function. The constraints (3) guarantee that each job is processed by exactly one machine.

We will also consider the tactical FIS problem with a robust chance constraint. Our goal is to find a minimum number of machines which can process all jobs under a prescribed probability that the schedule is feasible with respect to the worst-case distribution of the random delays:

$$
\begin{aligned}
\min_{x_{ji}, z_i} \ & \sum_{i=1}^{m} z_i \\
\text{s.t. } & x_{ji} \le z_i, \quad j = 1, \dots, n, \quad i = 1, \dots, m, \\
& \min_{P \in \mathcal{P}} P\left(\xi \in \Xi : \sum_{\{j:\, s_j \le t < f_j(\xi)\}} x_{ji} \le 1, \ t \in \mathcal{S}, \ i = 1, \dots, m \right) \\
& \ge 1 - \varepsilon, \\
& z_i \in \{0, 1\}, \quad i = 1, \dots, m, \\
& \text{constraints (2)} - \text{(4)}.
\end{aligned} \tag{5}
$$

Additional binary variables z_i are used to decide if machine i is used ($z_i = 1$) or it is left idle ($z_i = 0$) during the whole planning horizon. We assume that the number of machines m is sufficient for the problem to be feasible. A prescribed (risk) level $\varepsilon \in (0, 1)$ corresponds to the allowed probability of violating the schedule feasibility.

Like Branda et al. (2016) and Branda and Hájek (2017), we will consider a joint distribution of delays which follows an Archimedean copula. Let $F_j(x) = P(D_j(\xi) \le x)$ be the univariate cumulative distribution functions, and $\psi : [0, 1] \longrightarrow [0, \infty]$ be a generator of an Archimedean copula, i.e., a continuous strictly decreasing function satisfying $\psi(1) = 0$, $\lim_{x \to 0_+} \psi(x) = \infty$, see McNeil and Nešlehová (2009), Hering and Stadtmüller (2012) for details.

Assumption A1: The joint probability distribution of random delays follows an n-dimensional Archimedean copula function with a generator ψ, i.e.,

$$
P(D_1(\xi) \le x_1, \dots, D_n(\xi) \le x_n) = \psi^{-1}\left(\sum_{j=1}^{n} \psi(F_j(x_j)) \right), \tag{6}
$$

The second assumption introduces the structure of the cardinality constrained ambiguity set.

Assumption A2: For the stressing cdf $\tilde{F}_j$ it holds $\tilde{F}_j(x) < F_j(x), x \ge 0, \forall j$. We consider the cardinality constrained ambiguity set of the form

$$
\begin{aligned}
\mathcal{P} = \Big\{ & \Gamma \text{ marginal distributions } F_j \text{ are stressed to } \tilde{F}_j, \\
& n - \Gamma \text{ marginal distributions } F_j \text{ remain unchanged}, \\
& \text{joint distribution follows an Arch. copula} \\
& \text{with generator } \psi \text{ from Ass. A1} \Big\}
\end{aligned}
$$

with $0 < \Gamma \le n - m$.

Now, we would like to discuss the relevance of the assumptions and how the parameters can be obtained in a real application. We consider the gate assignment problem where the machines correspond to the gates and the jobs are the flights which must be served. Based on historical data, we can obtain the point estimates for the probability of no delay $p_j \in [0, 1]$ for the job j. The exponential distribution with parameter λ_j can be used to fit the delay length. Then, the marginal cumulative distribution function is

$$
F_j(x) = p_j + (1 - p_j)(1 - e^{-\lambda_j x}), \ x \ge 0. \tag{7}
$$

We can also get the confidence intervals for the parameters p_j, λ_j on which the stressing distribution $\tilde{F}_j$ can be based. When $\tilde{p}_j$ is a lower bound for the confidence interval and $\tilde{\lambda}_j$ is an upper bound, we can obtain $\tilde{F}_j(x)$ using formula (7). It can be observed that a proportion of flights is more delayed, e.g., based on unpredictable complications at the origins of the flights, which enables us to estimate the cardinality parameter Γ. There could also be some external conditions, mainly the weather around the considered airport, which can cause delays of all flights. This aspect is taken into account by the choice of the copula that is able to reproduce a positive dependence between the marginal delay distributions. As we will show in the numerical part, ignoring the dependence can lead to wrong conclusions, especially in the tactical FIS problem.

3. Robust operational problem

In this section, we focus on the robust operational FIS problem (1). We will show that, under Assumptions A1 and A2, it can be reformulated using the Conditional Value at Risk (CVaR) measure as a network flow problem with cost-CVaR objective function. The main advantage of this approach is that the resulting formulation is no longer a minimax problem, but a tractable minimization problem. Moreover, we will propose a decomposition algorithm running in polynomial time and discuss its further improvement using the golden-section search method.

3.1. Network flow reformulation

First, we introduce the Conditional Value at Risk (CVaR) measure briefly. Let Z be a random loss variable with S equiprobable realizations Z^s. Then CVaR can be defined using the minimization formula, cf. Rockafellar and Uryasev (2002),

$$
\mathrm{CVaR}_\alpha(Z) = \min_{\theta} \left\{ \theta + \frac{1}{(1-\alpha)S} \sum_{s=1}^{S} \max\{Z^s - \theta, 0\} \right\}. \tag{8}
$$

where the minimum is attained at a point of a bounded interval.

The network flow reformulation of the operational FIS problem is outlined in Fig. 1 where the flow corresponding to the assignment of 3 jobs to 2 machines is plotted by the colored edges. Each job is represented by two nodes which are connected by an edge with zero costs. Edges with positive costs connect end times with higher or equal starting times. The formal description of the network is given in the following proposition.

Proposition 3.1. *Under Assumptions A1 and A2, the robust operational FIS problem (1) can be reformulated as a network flow problem with cost–Conditional Value at Risk objective,*

$$
\begin{aligned}
\min_{y_{uv}} \ & \sum_{(u,v) \in \overline{E}} c_{uv} y_{uv} + \Gamma \ \mathrm{CVaR}_\alpha(Z(y)) \\
\text{s.t. } & \sum_{u:\, (v,u) \in E} y_{vu} - \sum_{u:\, (u,v) \in E} y_{uv} = d_v, \ v \in \mathcal{V}, \\
& y_{uv} \in \{0, 1\},
\end{aligned} \tag{9}
$$

where the network is defined as follows:

1. $2n + 2$ *vertices* $\mathcal{V}$: $\{0, s_1, f_1, \dots s_n, f_n, 2n + 1\}$; *vertices 0,* $2n + 1$ *correspond to the source and sink,*
2. *oriented edges E:* $\{0, s_j\}$, $\{s_j, f_j\}$, $\{f_k, s_j\}$ *if* $f_k \le s_j$, $\{f_j, 2n + 1\}$, $j, k = 1, \dots, n$,

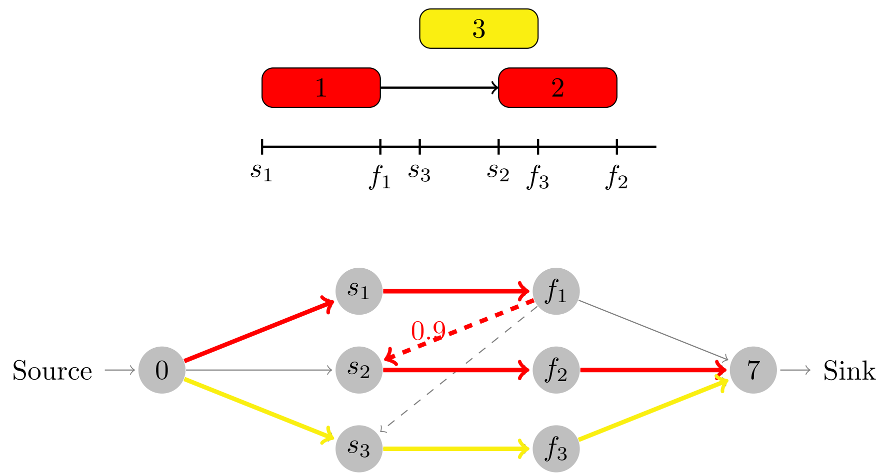

Fig. 1. Operational FIS: a schedule of 3 jobs on 2 machines and the corresponding network flow (bold lines identify the assignment of jobs to machines and their order).

3. *demands:* $d_0 = m,\ d_{2J+1} = -m,\ d_{s_j} = -1,\ d_{f_j} = 1,\ j = 1, \dots, n,$
4. *edges with positive costs* $\overline{E} \subset E$ *containing:* $\{f_k, s_j\}$ *if* $f_k \le s_j$*, i.e., vertices connecting finishing and starting times of the jobs that can be assigned to the same machine,*
5. *costs:* $c_{f_k,s_j} = \psi(P(D_k(\xi) \ge s_j - f_k)),\ \ \{f_k, s_j\} \in \overline{E},\ \ c_{uv} = 0$ *for* $(u, v) \in E \setminus \overline{E}$,
6. *stressed costs:* $\tilde{c}_{f_k,s_j} = \psi(\tilde{P}(D_k(\xi) \ge s_j - f_k)),\ (f_k, s_j) \in \overline{E}$,
7. *cost differences:* $\Delta_{u,v} = \tilde{c}_{u,v} - c_{u,v} > 0,\ (u, v) \in \overline{E}$,
8. *CVaR level:* $\alpha = 1 - \Gamma/|\overline{E}|$,
9. *loss random variable* $Z(y)$ *(dependent on decision variables* y_{uv}*) with equiprobable realizations* $\Delta_{uv}y_{uv}$, $(u, v) \in \overline{E}$.

The worst-case probability can be obtained by applying ψ^{-1} *to the optimum value of (9).*

The proof can be found in Appendix A.

The cost–CVaR network flow problem (9) can be reformulated using nonnegative decision variables as a (large) mixed-integer problem (17). However, the constraint matrix is not totally unimodular, and the optimum solution need not be integral if we relax $y_{uv} \in \{0, 1\}$. Thus the problem has to be solved using a MIP algorithm, which will be shown as inefficient in the numerical study. We will avoid dealing with the MIP problem with the aid of the decomposition proposed below.

We can compare our network flow reformulation with the others available in the literature for the deterministic FIS problems. Arkin and Silverberg (1987) dealt with the maximization of the weights of the selected jobs. For identical machines, they provided a network flow reformulation where the graph nodes correspond to the maximal cliques of the interval graph and the edges to the jobs. Bouzina and Emmons (1996) solved the same problem using a network where the nodes correspond to the jobs sorted according to the starting times with subsequent jobs connected by an arc with zero cost. The other arcs connect each job with the first one for which the intervals do not overlap. The arcs have capacity one and the costs correspond to the minus job weight. Mäkinen et al. (2017) maximized the minimum number of busy machines at each time. The nodes correspond to different starting and finishing times and the edges with capacities equal to the numbers of idle machines connect subsequent times. The jobs are represented by other edges with capacities 1 connecting the starting and finishing times. To summarize, all previous reformulations are focused on problems different from ours and do not enable us to assign penalties directly according to the subsequent jobs assigned to the same machine.

3.2. Algorithms based on decomposition

In this part, we will discuss two approaches to the solution based on the decomposition of the cost–CVaR network flow problem (9). It can be decomposed as the first-stage (master) problem

$$\min_{\theta} \Gamma\theta + \varphi(\theta) \text{ s.t. } \theta \in [0, \max_{(u,v)\in\overline{E}} \Delta_{uv}] \tag{10}$$

and the second-stage (slave) problem of the form

$$\begin{aligned} \varphi(\theta) = \min_{y_{uv}} & \sum_{(u,v)\in\overline{E}} c_{uv}y_{uv} + \sum_{(u,v)\in\overline{E}} \max\{\Delta_{uv}y_{uv} - \theta, 0\} \\ \text{s.t.} & \sum_{u:\ (v,u)\in E} y_{vu} - \sum_{u:\ (u,v)\in E} y_{uv} = d_v,\ \ v \in \mathcal{V}, \\ & y_{uv} \in \{0, 1\}. \end{aligned} \tag{11}$$

Note that the restriction on the variable θ in (10) follows from the properties of CVaR. We will call φ the second stage value function. We can show that, for a fixed θ, the slave problem (11) can be solved as a min-cost flow problem with relaxed binary variables. Using the trick suggested by Bertsimas and Sim (2003) for general robust combinatorial problems, if $y_{uv} \in \{0,\ 1\}$ and $\theta \in [0, \max_{(u,v)\in\overline{E}} \Delta_{uv}]$ is fixed, then we obtain the equality:

$$\max\{\Delta_{uv}y_{uv} - \theta, 0\} = \max\{\Delta_{uv} - \theta, 0\}y_{uv}.$$

We then get the min-cost flow problem with modified costs

$$\begin{aligned} \varphi(\theta) = \min_{y_{uv}} & \sum_{(u,v)\in\overline{E}} (c_{uv} + \max\{\Delta_{uv} - \theta, 0\})y_{uv} \\ \text{s.t.} & \sum_{i:\ (v,u)\in E} y_{vu} - \sum_{i:\ (u,v)\in E} y_{uv} = d_v,\ \ v \in \mathcal{V}, \\ & 0 \le y_{uv} \le 1. \end{aligned} \tag{12}$$

We can relax the binary restrictions to $y_{uv} \in [0,\ 1]$ because the constraint matrix is totally unimodular. Thus, the slave problem (11) can be efficiently solved in polynomial time. We will employ the golden-section search algorithm, cf. Bazaraa et al. (2006), to solve the master problem (10) based on a quick evaluation of $\varphi(\theta)$ in the particular points θ suggested by the algorithm. The required number of evaluations $\hat{n}$ of the second stage value function φ can be estimated as

$$(0.618)^{\hat{n}-1} \le \frac{\epsilon}{\max_{(u,v)\in\overline{E}} \Delta_{uv}}, \tag{13}$$

where ϵ is the length of the final interval, i.e., the prescribed precision with respect to the optimal solution. The steps are summarized in Algorithm 3.1. Convergence of the GSS algorithm, cf.

Bazaraa et al. (2006), Section 8.1, requires strict quasi-convexity of the minimized function which we cannot guarantee for this problem (10) in general. However, we will observe the convergence to a global minimum for all simulated instances in the numerical study. The situation is considered in detail in Appendix B.

Algorithm 3.1 Golden-section search.

Set the golden ratio to $\Phi := (\sqrt{5}-1)/2$
Set $lb := 0$ and $ub := \max_{(u,v)\in\bar{E}} \Delta_{uv}$
Set $\theta_1 = lb + (1-\Phi) * (ub - lb)$, $\theta_2 = lb + \Phi * (ub - lb)$
Evaluate $f_1 = \Gamma\theta_1 + \varphi(\theta_1)$, $f_2 = \Gamma\theta_2 + \varphi(\theta_2)$
while $ub - lb > \epsilon$ **do**
  if $f_1 < f_2$ **then**
    Set $ub := \theta_2$, $\theta_2 := \theta_1$, $\theta_1 = lb + (1-\Phi) * (ub - lb)$
  else
    Set $lb := \theta_1$, $\theta_1 := \theta_2$, $\theta_2 = lb + \Phi * (ub - lb)$
  end if
  Evaluate $f_1 = \Gamma\theta_1 + \varphi(\theta_1)$, $f_2 = \Gamma\theta_2 + \varphi(\theta_2)$
end while
if $f_1 < f_2$ **then**
  Return optimal solution θ_1, and optimal value f_1
else
  Return optimal solution θ_2, and optimal value f_2
end if

Bertsimas and Sim (2003) in their *Algorithm A* suggested solving the robust combinatorial problems by evaluating $\varphi(\theta)$ for all different values of θ contained in $\{\Delta_{uv},\ (u,v)\in\bar{E}\}$ and by taking the minimum over the objective values, i.e., in our case

$$\min_{\theta\in\{\Delta_{uv},\ (u,v)\in\bar{E}\}} \Gamma\theta + \varphi(\theta). \tag{14}$$

Since both previous approaches lead to a polynomial number of operations, we will compare their performance in a numerical study. Note that Algorithm B proposed by Bertsimas and Sim (2003) for the robust network flow problems cannot be used for our robust FIS problem since the authors considered the problems with real (nonintegral) flows, which would correspond to the LP relaxation of problem (17) without the total unimodularity of the constraint matrix.

4. Robust tactical problem

In this section, we focus on the robust tactical FIS problem (5), which was formulated as a chance constrained problem. Although the chance constrained problems are highly demanding in general, see Prékopa (2003) and Shapiro et al. (2009), we will show that we can use the results obtained in the previous section to solve our tactical problem. The following corollary is based on an observation that by increasing the number of available machines, the best attainable schedule probability cannot be decreased.

Corollary 4.1. *If, for a certain number of machines $\hat{m}$, the optimal probability in the robust operational problem (1) is greater than or equal to $1-\varepsilon$, and for $\hat{m}-1$ it is smaller than $1-\varepsilon$, then such $\hat{m}$ is the optimum value for the tactical problem (5).*

First, we must find the lower and upper numbers of machines which are needed to start the binary search. A minimum number of machines necessary to process all jobs can be obtained using the left-edge algorithm running in $O(n\log n)$ time, cf. Gupta et al. (1979), which first sorts the jobs according to their starting times and then assigns the jobs sequentially to the first available machine. Usually this algorithm leads to schedules with very low probability. The upper number of machines must lead to a schedule with a probability value higher than $1-\varepsilon$. A trivial upper bound with 100 percent probability corresponds to the case in which each job is assigned to a single machine. Since we do not see an easy way of getting a tight upper bound on the number of machines for which the probability reaches at least $1-\varepsilon$, we will increase the number of machines twice at each iteration until reaching a probability of at least $1-\varepsilon$. When we get to the upper bound, we can start the regular binary search with performance $O(\log m)$. Our modified binary search is summarized in Algorithm 4.1.

Algorithm 4.1 Modified binary search.

Run the left-edge algorithm to obtain the minimum number of machines $blm := m$ and compute the corresponding probability $blr := prob$.
Set $bur := \infty$, $bum := \infty$.
while $bum - blm > 1$ **do**
  if $bur = \infty$ **then**
    $m := 2m$
  else
    $m := round((blm + bum)/2)$
  end if
  Solve the *robust operational FIS* (1) given m machines using Algorithm 3.1 and obtain probability *prob*
  if $prob >= 1-\varepsilon$ **then**
    $bur := prob$,
    $bum := m$
  else
    $blr := prob$
    $blm := m$
  end if
end while
Return $\hat{m} := bum$

5. Numerical study

In this section, we compare the computational performance of the introduced approaches to the solution on simulated FIS instances. Three sets of test instances per 10 problems are considered, where the job lengths and the breaks between the jobs are simulated using the exponential distribution. We consider one parameter value $\mu_2 = 0.2$ for the job length, whereas three different values $\mu_1 \in \{0.2,\ 0.5,\ 0.8\}$ are taken for the breaks. The expected length of a job is 5 units of time, and the expectations of breaks between the jobs are 5, 2, and 1.25 units, i.e., we increase the urgency by increasing the parameter μ_1. The test instances were originally simulated to 10 machines with 10 jobs assigned to each of them. We set the probability that a job is finished in time to $p_j = 0.95,\ j = 1,\dots,n$, i.e., there is a high probability that each job is finished in time f_j. We will stress the distribution of delays (7) by decreasing the probability p_j by a random number simulated from the uniform distribution on the interval (0, 0.15) leading to $\tilde{p}_j < p_j$. The way of stressing the parameter p_j has no practical meaning; it should just simulate a different change for different jobs which can be expected in practice. The value of parameter $\lambda_j = 0.5$ is left unchanged. According to formula (7), this approach leads to $\tilde{F}_j(x) < F_j(x)$ for all $x \geq 0$, as we require in Assumption A2. We consider three possible values of the cardinality parameter $\Gamma \in \{10, 20, 40\}$, i.e., we expect that 10, 20, or 40 percent of jobs (out of 100) can be delayed with a higher probability $\tilde{p}_j$. All computations including simulations, preprocessing and optimization were performed using Matlab R2016b on PC with Intel Xeon CPU E3-1220 3.00 GHz, 32 GB RAM, and Windows 7 64 bit system.

5.1. Robust operational problem

In this part, we employ the introduced approaches for solving the robust operational problem. In particular, we compare compu-

Table 1
Operational FIS: summary results (each column contains an average of over 10 instances).

	$\mu_1 = 0.2,\ \mu_2 = 0.2$			$\mu_1 = 0.5,\ \mu_2 = 0.2$			$\mu_1 = 0.8,\ \mu_2 = 0.2$		
$\lvert E\rvert$	4869.0	4869.0	4869.0	4711.5	4711.5	4711.5	4642.8	4642.8	4642.8
$\lvert\bar{E}\rvert$	2623.6	2623.6	2623.6	3258.1	3258.1	3258.1	3428.3	3428.3	3428.3
Γ	10	20	40	10	20	40	10	20	40
Mixed-integer programming formulation									
φ^{MIP}	0.870	0.784	0.665	0.644	0.477	0.297	0.546	0.370	0.197
t^{MIP}	6:12	6:12	6:11	6:05	6:05	6:05	6:02	6:02	6:02
abs. gap	−0.014	−0.033	−0.066	−0.023	−0.054	−0.077	−0.031	−0.061	−0.071
rel. gap	−1.6%	−4.1%	−9.0%	−3.4%	−10.2%	−20.5%	−5.4%	−14.1%	−26.5%
Bertsimas–Sim algorithm									
φ^{B-S}	0.884	0.817	0.730	0.667	0.531	0.374	0.577	0.431	0.269
t^{B-S}	7:21	7:19	7:19	7:32	7:32	7:34	7:30	7:33	7:31
iter.	535.4	535.4	535.4	624.6	624.6	624.6	660.2	660.2	660.2
Golden-section search algorithm									
φ^{GSS}	0.884	0.817	0.730	0.667	0.531	0.374	0.577	0.431	0.269
t^{GSS}	0:28	0:28	0:27	0:26	0:25	0:24	0:25	0:24	0:22
iter.	16	16	16	16	16	16	16	16	16

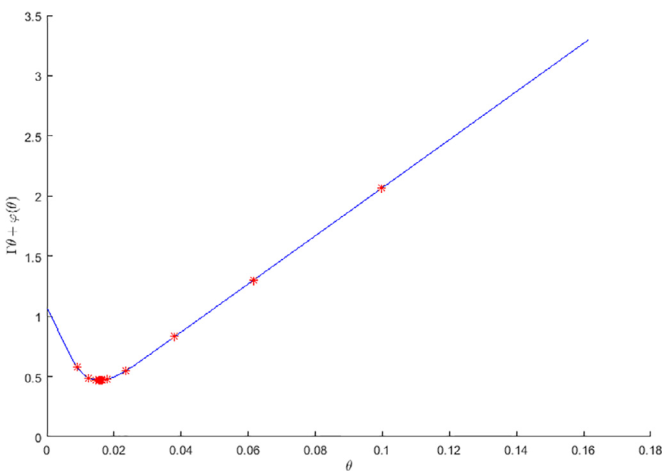

Fig. 2. Operational problem ($\Gamma = 20,\ \mu_1 = 0.5,\ \mu_2 = 0.2$): objective function of the master problem (10) evaluated by the B–S algorithm and iterations of the GSS algorithm (red stars). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

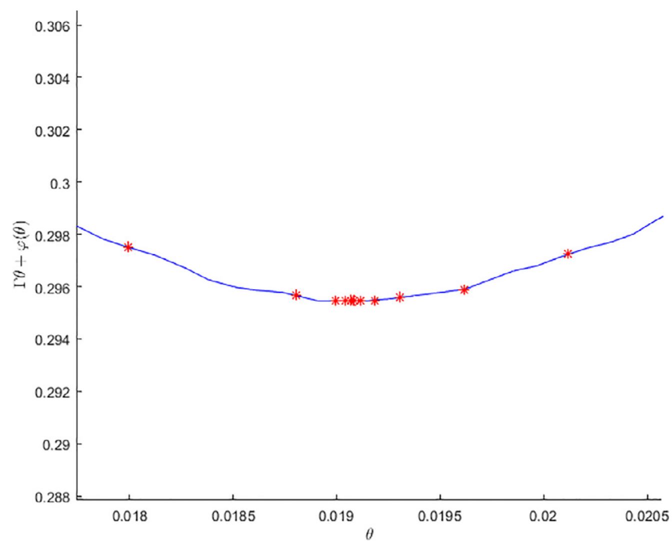

Fig. 3. Operational problem ($\Gamma = 20,\ \mu_1 = 0.5,\ \mu_2 = 0.2$): objective function of the master problem (10) – detailed view around the minimal point.

tational performance of the mixed-integer reformulation (17), evaluation algorithm (14) proposed by Bertsimas and Sim (2003) (B–S algorithm), and our approach based on the golden-section search method (GSS algorithm 3.1).

Table 1 shows a summary for all instances and choices of the parameter $\Gamma \in \{10,\ 20,\ 40\}$ and $m = 20$ machines. It contains the number of all edges $|E|$ and the number of edges with positive costs $|\bar{E}|$, the lowest value of the objective $\varphi^{\bullet}$, and the computational time $t^{\bullet}$. Moreover, we report tssshe number of evaluations of the second stage value function φ for the evaluation algorithm and the golden-section search. For the evaluation algorithm, this does not correspond to the number of edges with positive costs $|\bar{E}|$ because we selected the same tolerance $\epsilon = 0.001$ here as for the Algorithm 3.1 and we evaluate function φ only at the points which differ at least by ϵ. The running times of these algorithms t^{B-S}, t^{GSS} are quite stable with respect to the problem parameters. The average time is around 7 min and 30 s for the B–S algorithm, whereas the GSS algorithm found the optimal solutions in less than 30 seconds on average. The MIP approach was not able to provide an optimal solution in the time limit of 300 s for computations and similar time for preprocessing. The absolute and relative gap of the objective values reported for the MIP approach are computed with respect to the optimal values found by the B–S and GSS algorithms. We can see from Table 1 that the relative gap is quite unstable; however, on average it decreases with the increase in the urgency parameter μ_1 and the cardinality parameter Γ.

Fig. 2 plots the objective function $\Gamma\theta + \varphi(\theta)$ of one particular master problem (10) for all values evaluated by the B–S algorithm. The points investigated by the GSS algorithm are marked by the red stars. A detailed view on the shape and evaluated points around the optimum can be seen in Fig. 3. It is obvious that the objective function is not convex, but it is strictly quasi-convex.

5.2. Robust tactical problem

In this part, we will focus on the tactical problem (5) and apply the Algorithm 4.1 (Modified binary search). We will investigate the influence of dependence on the optimum number of machines. We employ Gumbel's copula with the generating function $\psi(x) = (-\log x)^{\nu},\ \nu \in [1, \infty)$, We consider not only the independent delay distributions under $\nu = 1$, but also the dependence with the parameter choices $\nu = 1.5$ and $\nu = 2$ corresponding to the Kendall's tau[1] $\tau = 0.5$ and $\tau = 2/3$.

[1] Kendal's τ is a nonparametric rank correlation coefficient which has properties similar to those of the Pearson coefficient, i.e., the positive values close to one correspond to a strong positive dependence, whereas the negative values close to −1 show a negative dependence. For the Gumbel copula, it takes nonnegative values depending on the parameter $\tau = 1 - \frac{1}{\nu}$. Thus, this copula is suitable to model positive tail dependencies observable, e.g., during a crisis.

Table 2
Tactical FIS: summary results (each column contains an average of over 10 instances).

	$\mu_1 = 0.2,\ \mu_2 = 0.2$			$\mu_1 = 0.5,\ \mu_2 = 0.2$			$\mu_1 = 0.8,\ \mu_2 = 0.2$		
$\lvert E\rvert$	4869	4869	4869	4711.5	4711.5	4711.5	4642.8	4642.8	4642.8
$\lvert\bar{E}\rvert$	2623.6	2623.6	2623.6	3258.1	3258.1	3258.1	3428.3	3428.3	3428.3
Γ	10	20	40	10	20	40	10	20	40
					$\nu = 1.0$				
prob.	0.955	0.955	0.955	0.953	0.953	0.954	0.953	0.953	0.953
$\hat{m}$	24.6	27.4	29.4	33.9	37.4	40.4	36.6	40.8	44.3
time	3:10	3:11	3:05	2:28	3:04	3:15	3:03	3:16	3:25
iter.	7	7.2	7.4	7.4	7.7	8.4	7.5	8.4	8.9
					$\nu = 1.5$				
prob.	0.955	0.955	0.955	0.953	0.953	0.955	0.954	0.953	0.954
$\hat{m}$	20.8	22.4	23.6	28.2	30.6	32.7	30.8	33.3	35.7
time	3:22	3:14	3:15	3:09	3:04	3:01	2:57	2:50	2:52
iter.	7.2	7	7	7.6	7.4	7.4	7.6	7.3	7.5
					$\nu = 2.0$				
prob.	0.956	0.956	0.955	0.954	0.953	0.953	0.953	0.954	0.953
$\hat{m}$	19	20	20.8	25.6	27.2	28.4	27.7	29.8	31.1
time	3:15	3:23	3:23	2:48	3:07	3:04	2:52	2:54	2:56
iter.	7	7.2	7.2	7.4	7.6	7.5	7.4	7.5	7.6

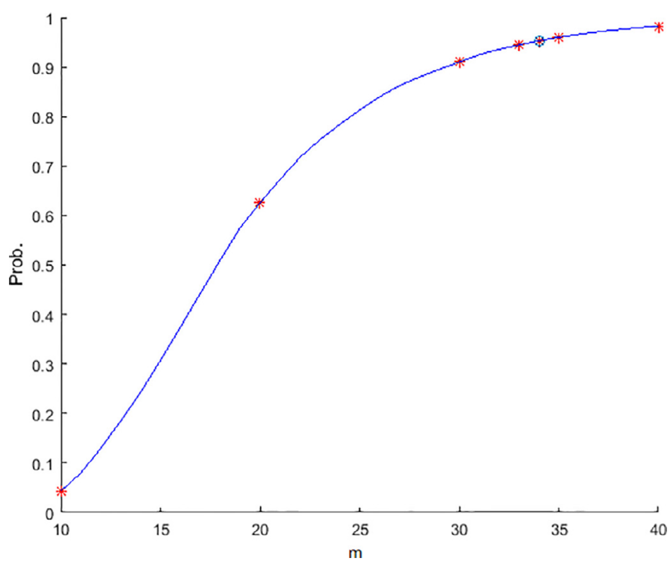

Fig. 4. Tactical problem ($\Gamma = 20,\ \varepsilon = 0.05,\ \mu_1 = 0.5,\ \mu_2 = 0.2$): efficient frontier and points evaluated by the Modified binary search (Alg. 4.1). (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

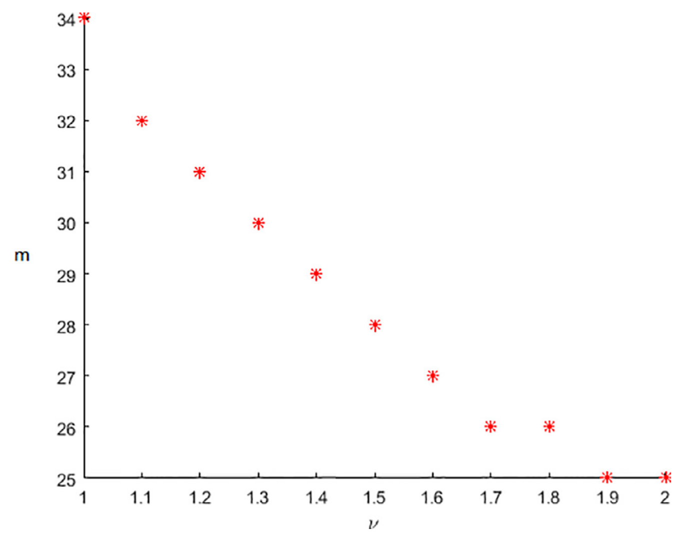

Fig. 5. Tactical problem ($\Gamma = 20,\ \varepsilon = 0.05,\ \mu_1 = 0.5,\ \mu_2 = 0.2$): optimal number of machines depending on the parameter ν of Gumbel's copula.

Table 2 shows a summary for all instances considered in the previous subsection under the choice $\varepsilon = 0.05$. As we can expect, the optimum number of machines $\hat{m}$ increases with the urgency parameter μ_1 and the cardinality parameter Γ, whereas it decreases with the parameter of Gumbel's copula ν. This dependence of the optimum number of machines on the parameter $\nu \in [1, 2]$ of Gumbel's copula can be observed in Fig. 5. We can see that, for higher values of ν, the optimum number of machines is not changing rapidly, but by changing $\nu = 1$ to $\nu = 1.1$ (small dependency $\tau = 0.09$) the optimum number of machines was decreased from 34 to 32. Maybe surprisingly, the number of iterations of the modified binary search is also increased with parameters μ_1 and Γ. However, the number of iterations was always smaller than or equal to 10.

Fig. 4 shows the whole efficient frontier connecting the highest attainable probability (vertical axis) for a given number of machines (horizontal axis). The red stars represent the points which were evaluated by the modified binary search algorithm. If a decision maker prefers knowing the whole frontier, the optimum number of machines for a given level ε can be obtained easily, but the binary-search algorithm can reduce the computational effort significantly.

6. Conclusions

We have dealt with the fixed interval scheduling problems on identical machines under uncertain delays in the processing times. We have focused on the distributionally robust problems with the cardinality constrained ambiguity set where the joint distribution follows an Archimedean copula and a proportion of the marginal distributions is stressed. We have shown that the robust operational problem can be reformulated as a robust network flow problem with a cost–CVaR objective. We have proposed a decomposition algorithm where the slave problem is solved as a network flow problem with the relaxed binary variables, and the master problem is approached by the golden-section search algorithm. This algorithm has been shown to be effective in the numerical study compared with the mixed-integer solver and the evaluation algorithm proposed by Bertsimas and Sim (2003). We have proposed an extended binary search for the robust tactical problem. This algorithm solves the robust operational problems for the numbers of machines given during the iterations of the binary search. Moreover, we have shown that the delay dependence influences the optimum number of machines. Future research will be devoted to the robust problems with several machine classes.

Acknowledgements

The financial support of the Czech Science Foundation under Grant No. P402/12/G097 (DYME) is acknowledged. The author would like to express his gratitude to the anonymous referees for their valuable comments and suggestions.

Appendix A. Proof of Proposition 3.1

Proof. To express the worst-case probability in the problem (1), we employ new binary variables $\tilde{z}_j \in \{0, 1\}$, where $\tilde{z}_j = 1$ if the marginal distribution of $D_j(\xi)$ is stressed, and $\tilde{z}_j = 0$ if it is left unchanged. According to Assumption A2, it holds that $\sum_{j=1}^n \tilde{z}_j = \Gamma$, i.e., Γ marginals are changed. Then, we can control the number of marginal distributions which are stressed in the joint probability that can be expressed as

$$C\Big((1-\tilde{z}_1)P_1(D_1(\xi) \le s_{k(1)} - f_1) + \tilde{z}_1\tilde{P}_1(D_1(\xi) \le s_{k(1)} - f_1), \ldots$$
$$\ldots, (1-\tilde{z}_n)P_j(D_n(\xi) \le s_{k(n)} - f_n) + \tilde{z}_n\tilde{P}_n(D_n(\xi) \le s_{k(n)} - f_n)\Big)$$
$$= \psi^{-1}\left(\sum_{j=1}^n \psi\Big((1-\tilde{z}_j)P_j(D_j(\xi) \le s_{k(j)} - f_j) + \tilde{z}_j\tilde{P}_j(D_j(\xi) \le s_{k(j)} - f_j)\Big)\right)$$
$$= \psi^{-1}\left(\sum_{j=1}^n (1-\tilde{z}_j)\psi\Big(P_j(D_j(\xi) \le s_{k(j)} - f_j)\Big) + \tilde{z}_j\psi\Big(\tilde{P}_j(D_j(\xi) \le s_{k(j)} - f_j)\Big)\right),$$

where the first equality holds for any Archimedean copula and, in the second equality, we interchanged selection of the marginal probability and (monotonous) inverse of the generator ψ. Taking $\tilde{z}_j = 1$ for job j which is assigned to the last position on a machine, is never optimal, because its delay does not contribute to the probability. Thus, we can consider only jobs with the followers.

Now, we can move to the network flow formulation. To simplify the explanation, we assume that all flows y_{uv} are contained in the following set of feasible binary flows

$$\left\{ y_{uv} \in \{0,1\} : \ (u,v) \in E, \ \sum_{u:\,(v,u)\in E} y_{vu} - \sum_{u:\,(u,v)\in E} y_{uv} = d_v, \ v \in \mathcal{V} \right\}.$$

Based on the previous discussion, we can consider only edges (u, v) from $\bar{E}$ corresponding to the connections between two jobs with positive costs. Since y_{uv} is equal to one for each $u \in \mathcal{V}$ and at most one v (possibly there is a unit flow $y_{u,2n+1} = 1$ without any cost if the job corresponding to the node u is the last one processed by a machine), i.e.,

$$\sum_{v:\,(u,v)\in\bar{E}} y_{uv} \le 1, \ u \in \mathcal{V},$$

then at most one coefficient c_{uv} can be worsened to $\tilde{c}_{uv}$ for each node u. Thus, we may employ binary variables $\tilde{z}_{uv}$ which have the meanings similar to the variable $\tilde{z}_j$ used above; i.e., they identify the edges $(u, v) \in \bar{E}$ with the stressed costs. Since now, we assume that the new binary variables $\tilde{z}_{uv}$ are contained in the set

$$\left\{ \tilde{z}_{uv} \in \{0,1\} : \ \sum_{(u,v)\in\bar{E}} \tilde{z}_{uv} = \Gamma \right\}.$$

The maximization of the worst-case probability can be expressed as

$$\max_{y_{uv}} \min_{\tilde{z}_{uv}} \psi^{-1}\left(\sum_{(u,v)\in\bar{E}} ((1-\tilde{z}_{uv})c_{uv} + \tilde{z}_{uv}\tilde{c}_{uv})y_{uv}\right)$$
$$= \psi^{-1}\left(\min_{y_{uv}} \max_{\tilde{z}_{uv}} \sum_{(u,v)\in\bar{E}} ((1-\tilde{z}_{uv})c_{uv} + \tilde{z}_{uv}\tilde{c}_{uv})y_{uv}\right)$$
$$= \psi^{-1}\left(\min_{y_{uv}}\left(\sum_{(u,v)\in\bar{E}} c_{uv}y_{uv} + \max_{\tilde{z}_{uv}} \sum_{(u,v)\in\bar{E}} \Delta_{uv}\tilde{z}_{uv}y_{uv}\right)\right), \tag{15}$$

where we used the monotonicity of ψ^{-1} and that the edge contribution to the probability is equal to

$$(1-\tilde{z}_{uv})c_{uv} + \tilde{z}_{uv}\tilde{c}_{uv} = c_{uv} + \tilde{z}_{uv}(\tilde{c}_{uv} - c_{uv}) = c_{uv} + \tilde{z}_{uv}\Delta_{uv}.$$

Thus, at most one Δ_{uv} is considered for a node u depending on the value of $\tilde{z}_{uv}$.

The inner maximization problem is looking for the highest Γ values among $\Delta_{uv}y_{uv}$; thus it can be rewritten with the relaxed decision variables $\tilde{z}_{uv}$ for fixed y_{uv} as the linear program

$$\begin{aligned} \max_{\tilde{z}_{uv}} \ & \sum_{(u,v)\in\bar{E}} (\Delta_{uv}y_{uv})\tilde{z}_{uv} \\ \text{s.t.} \ & \sum_{(u,v)\in\bar{E}} \tilde{z}_{uv} = \Gamma, \\ & 0 \le \tilde{z}_{uv} \le 1. \end{aligned} \tag{16}$$

Applying the linear programming duality to the maximization problem, as proposed by Bertsimas and Sim (2003), we can obtain a minimization problem in the form

$$\begin{aligned} \min_{y_{uv},\theta,p_{uv}} \ & \sum_{(u,v)\in\bar{E}} c_{uv}y_{uv} + \Gamma\theta + \sum_{(u,v)\in\bar{E}} p_{uv} \\ \text{s.t.} \ & \sum_{u:\,(v,u)\in E} y_{vu} - \sum_{u:\,(u,v)\in E} y_{uv} = d_v, \ v \in \mathcal{V}, \\ & p_{uv} \ge \Delta_{uv}y_{uv} - \theta, \\ & y_{uv} \in \{0,1\}, \ p_{uv} \ge 0. \end{aligned} \tag{17}$$

We avoid to the nonnegative variables p_{uv} using the positive parts in the objective function and get the equivalent problem:

$$\begin{aligned} \min_{y_{uv},\theta} \ & \sum_{(u,v)\in\bar{E}} c_{uv}y_{uv} + \Gamma\theta + \sum_{(u,v)\in\bar{E}} \max\{\Delta_{uv}y_{uv} - \theta, 0\} \\ \text{s.t.} \ & \sum_{u:\,(v,u)\in E} y_{vu} - \sum_{u:\,(u,v)\in E} y_{uv} = d_v, \ v \in \mathcal{V}, \\ & y_{uv} \in \{0,1\}. \end{aligned} \tag{18}$$

By rewriting the objective function as

$$\sum_{(u,v)\in\bar{E}} c_{uv}y_{uv} + \Gamma\left(\theta + \frac{|\bar{E}|}{\Gamma} \sum_{(u,v)\in\bar{E}} \frac{1}{|\bar{E}|} \max\{\Delta_{uv}y_{uv} - \theta, 0\}\right) \tag{19}$$

and comparing it with the minimization formula for CVaR (8), the proposition follows. □

Appendix B. Convergence of the GSS algorithm

We discuss the convergence of the golden-section search algorithm for the problem (10). In Bazaraa et al. (2006), Section 8.1, the convergence is proven under the condition that the minimized function is strictly quasi-convex[2]. Since the objective is a function of one real variable θ, we can characterize the strict quasi-convexity using its local slopes. We can obtain the local slope of

[2] Let $f : S \to \mathbb{R}$, where S is a nonempty convex set in $\mathbb{R}^n$. The function f is said to be strictly quasi-convex if for each $x_1, x_2 \in S$ with $f(x_1) \ne f(x_2)$ and $\lambda \in (0, 1)$, we have

$$f(\lambda x_1 + (1-\lambda)x_2) < \max\{f(x_1), f(x_2)\}.$$

the function φ at any $\theta \in [0, \max_{(u,v)\in\overline{E}} \Delta_{uv}]$ as

$$\text{slope}_{\varphi}(\theta) = - \sum_{(u,v)\in\overline{E},\ \Delta_{uv}>\theta} \hat{y}_{uv} \tag{20}$$

where $\hat{y}_{uv}$ is an optimal solution of (12) for such θ. Then, the objective function $\Gamma\theta + \varphi(\theta)$ is strictly quasi-convex if there are no points $\theta_1 < \theta_2 < \theta_3$ such that

$$-\text{slope}_{\varphi}(\theta_1) > \Gamma,\ \ -\text{slope}_{\varphi}(\theta_2) \le \Gamma,\ \ -\text{slope}_{\varphi}(\theta_3) > \Gamma,$$

or

$$-\text{slope}_{\varphi}(\theta_1) < \Gamma,\ \ -\text{slope}_{\varphi}(\theta_2) \ge \Gamma,\ \ -\text{slope}_{\varphi}(\theta_3) < \Gamma.$$

For instance, the first set of conditions means that there are intervals

- containing θ_1 where the objective function $\Gamma\theta + \varphi(\theta)$ is decreasing,
- containing θ_2 where it is nondecreasing,
- containing θ_3 where it is again decreasing.

Although we cannot guarantee strict quasi-convexity of the objective in general, we did not observe its violation in the numerical study; and the GSS algorithm always converges to the optimal solution obtained by the B–S algorithm where the convergence is guaranteed.

References

Angelelli, E., Bianchessi, N., Filippi, C., 2014. Optimal interval scheduling with a resource constraint. Comput. Oper. Res. 51, 268–281. doi:10.1016/j.cor.2014.06.002.

Angelelli, E., Filippi, C., 2011. On the complexity of interval scheduling with a resource constraint. Theor. Comput. Sci. 412 (29), 3650–3657. doi:10.1016/j.tcs.2011.03.025.

Arkin, E.M., Silverberg, E.B., 1987. Scheduling jobs with fixed start and end times. Discret. Appl. Math. 18 (1), 1–8. doi:10.1016/0166-218X(87)90037-0.

Bazaraa, M.S., Sherali, H.D., Shetty, C.M., 2006. Nonlinear Programming: Theory and Algorithms. Wiley-Interscience.

Bertsimas, D., Sim, M., 2003. Robust discrete optimization and network flows. Math ProgR. 98 (1), 49–71. doi:10.1007/s10107-003-0396-4.

Bertsimas, D., Sim, M., 2004. The price of robustness. Oper. Res. 52 (1), 35–53. doi:10.1287/opre.1030.0065.

Bouzina, K.I., Emmons, H., 1996. Interval scheduling on identical machines. J. Global Optim. 9 (3), 379–393. doi:10.1007/BF00121680.

Branda, M., Hájek, Š., 2017. Flow-based formulations for operational fixed interval scheduling problems with random delays. Comput. Manag. Sci. 14 (1), 161–177. doi:10.1007/s10287-016-0262-5.

Branda, M., Novotný, J., Olstad, A., 2016. Fixed interval scheduling under uncertainty – a tabu search algorithm for an extended robust coloring formulation. Comput. Ind. Eng. 93, 45–54. doi:10.1016/j.cie.2015.12.021.

Branda, M., Novotný, J., Olstad, A., Popela, P., 2015. A note on fixed interval scheduling with stochastic elements. Inf. Bull. Czech Stat. Soc. 26 (4), 1–7.

Chang, Z., Song, S., Zhang, Y., Ding, J.-Y., Zhang, R., Chiong, R., 2017. Distributionally robust single machine scheduling with risk aversion. Eur. J. Oper. Res. 256 (1), 261–274. doi:10.1016/j.ejor.2016.06.025.

Eliiyi, D.T., 2013. Integrating tactical and operational decisions in fixed job scheduling. Eng. Optim. 45 (12), 1449–1467. doi:10.1080/0305215X.2012.743532.

Gupta, U.I., Lee, D.T., Leung, J.Y.T., 1979. An optimal solution for the channel-assignment problem. IEEE Trans. Comput. C-28 (11), 807–810. doi:10.1109/TC.1979.1675260.

Hanasusanto, G.A., Roitch, V., Kuhn, D., Wiesemann, W., 2015. A distributionally robust perspective on uncertainty quantification and chance constrained programming. Math Progr. 151 (1), 35–62. doi:10.1007/s10107-015-0896-z.

Hering, C., Stadtmüller, U., 2012. Estimating archimedean copulas in high dimensions. Scand. J. Stat. 39 (3), 461–479. doi:10.1111/j.1467-9469.2012.00803.x.

Kolen, A.W., Lenstra, J.K., Papadimitriou, C.H., Spieksma, F.C., 2007. Interval scheduling: a survey. Nav. Res. Logist. 54 (5), 530–543.

Kovalyov, M.Y., Ng, C., Cheng, T.E., 2007. Fixed interval scheduling: models, applications, computational complexity and algorithms. Eur. J. Oper. Res. 178 (2), 331–342. doi:10.1016/j.ejor.2006.01.049.

Kroon, L.G., Salomon, M., Wassenhove, L.N.V., 1995. Exact and approximation algorithms for the operational fixed interval scheduling problem. Eur. J. Oper. Res. 82 (1), 190–205. doi:10.1016/0377-2217(93)E0335-U.

Kroon, L.G., Salomon, M., Wassenhove, L.N.V., 1997. Exact and approximation algorithms for the tactical fixed interval scheduling problem. Oper. Res. 45 (4), 624–638.

Lin, S.-W., Ying, K.-C., 2014. Minimizing shifts for personnel task scheduling problems: a three-phase algorithm. Eur. J. Oper. Res. 237 (1), 323–334. doi:10.1016/j.ejor.2014.01.035.

McNeil, A.J., Nešlehová, J., 2009. Multivariate archimedean copulas, d-monotone functions and l_1-noram symmetric distributions. Ann. Stat. 37 (5B), 3059–3097. doi:10.1214/07-AOS556.

Mäkinen, V., Staneva, V., Tomescu, A.I., Valenzuela, D., Wilzbach, S., 2017. Interval scheduling maximizing minimum coverage. Discret. Appl. Math. 225, 130–135. doi:10.1016/j.dam.2016.08.021.

Ng, C.T., Cheng, T.C.E., Bandalouski, A.M., Kovalyov, M.Y., Lam, S.S., 2014. A graph-theoretic approach to interval scheduling. J. Oper. Res. Soc. 65 (10), 1571–1579. doi:10.1057/jors.2013.109.

Prékopa, A., 2003. Probabilistic Programming. In: Ruszczyński, A., Shapiro, A. (Eds.), Stochastic Programming. In: Handbooks in Operations Research and Management Science, 10. Elsevier, pp. 267–351. doi:10.1016/S0927-0507(03)10005-9.

Rockafellar, R., Uryasev, S., 2002. Conditional value-at-risk for general loss distributions. J. Bank. Financ. 26 (7), 1443–1471. doi:10.1016/S0378-4266(02)00271-6.

Shapiro, A., Dentcheva, D., Ruszczyński, A., 2009. Lectures on Stochastic Programming. MOS-SIAM Series on Optimization. SIAM–Society for Industrial and Applied Mathematics, Philadelphia.

Smet, P., Wauters, T., Mihaylov, M., Berghe, G.V., 2014. The shift minimisation personnel task scheduling problem: a new hybrid approach and computational insights. Omega (Westport) 46, 64–73. doi:10.1016/j.omega.2014.02.003.

Wiesemann, W., Kuhn, D., Rustem, B., 2012. Multi-resource allocation in stochastic project scheduling. Ann. Oper. Res. 193 (1), 193–220. doi:10.1007/s10479-008-0486-z.

Žáčková, J., 1966. On minimax solutions of stochastic linear programming problems. Časopis pro pěstování matematiky 91 (4), 423–430.

Zhou, S., Zhang, X., Chen, B., van de Velde, S., 2014. Tactical fixed job scheduling with spread-time constraints. Comput. Oper. Res. 47, 53–60. doi:10.1016/j.cor.2014.02.001.

Zymler, S., Kuhn, D., Rustem, B., 2013. Distributionally robust joint chance constraints with second-order moment information. Math Progr. 137 (1–2), 167–198. doi:10.1007/s10107-011-0494-7.