S©MR ISSN 1813-3304

СИБИРСКИЕ ЭЛЕКТРОННЫЕ
МАТЕМАТИЧЕСКИЕ ИЗВЕСТИЯ

Siberian Electronic Mathematical Reports
http://semr.math.nsc.ru

Том 19, №2, стр. 548–561 (2022) УДК 519.854.2
DOI 10.33048/semi.2022.19.046 MSC 90B35

Special issue: International conference «Mathematical Optimization Theory and Operations Research» (MOTOR 2022), July 2–6, 2022, Petrozavodsk, Republic of Karelia, Russia

ON THE TWO-MACHINE ROUTING OPEN SHOP ON A TREE WITH PREEMPTION ALLOWED

P.M. AGZYAMOVA, I.D. CHERNYKH

ABSTRACT. The routing open shop problem is a natural generalization of the metric TSP and a classical open shop scheduling problem. Jobs are located at the nodes of a given transportation network, and mobile machines have to perform operations on those jobs while traveling over the edges. Machines are obligated to return to the initial location after completing all operations. The goal is to minimize the makespan. We consider the two-machine routing open shop on a tree with preemption in a general setting, where travel times are machine- and direction-dependent. For this problem we describe a wide polynomially solvable special case, for which the optimal makespan is guaranteed to coincide with the standard lower bound. To that end, we introduce a new problem setting with restricted preemption.

Keywords: shop scheduling, routing open shop, restricted preemption, individual travel times, asymmetric transportation network, polynomially solvable cases, standard lower bound.

AGZYAMOVA, P.M., CHERNYKH, I.D., ON THE TWO-MACHINE ROUTING OPEN SHOP ON A TREE WITH PREEMPTION ALLOWED.

This research was carried out within the framework of the state contract of the Sobolev Institute of Mathematics (project FWNF-2022-0019), and supported by the Russian Foundation for Basic Research, projects 20-01-00045 and 20-07-00458.

Received April, 21, 2022, published August, 26, 2022.

1. Introduction

The object of investigation of this paper is so-called *routing open shop problem* with preemption allowed, which is a natural generalization of a well-known metric traveling salesman problem [11] and preemptive open shop scheduling problem [13]. The metric TSP hardly needs a special introduction. The open shop problem can be described as follows. Given a set of *jobs* $\mathcal{J} = \{J_1, \ldots, J_n\}$, one need to construct a schedule of processing of each job J_j by every *machine* from a given set $\mathcal{M} = \{M_1, \ldots, M_m\}$, *i.e.* to specify a processing interval $[s_{ji}, c_{ji}]$ for each *operation* O_{ji} (which is essentially a pair $\langle J_j, M_i\rangle$). The length of this interval is called *processing time* $p_{ji} = c_{ji} - s_{ji}$, those processing times are given *a priori*. Processing intervals of any two operations involving the same job or the same machine cannot overlap. The goal is to construct a feasible schedule minimizing the latest completion time $C_{\max} = \max c_{ji}$, also known as the *makespan*.

Following the traditional three-field notation for scheduling problems (see *e.g.* [19]), we denote the open shop problem with m machines as $Om||C_{\max}$. Notation $O||C_{\max}$ is used for the case when the number of machines is not bounded by any constant and is a part of the input. The two-machine problem $O2||C_{\max}$ is solvable in linear time $O(n)$ by different known algorithms, developed since 1976 [13, 14, 20, 24, 25]. Moreover, the optimal makespan $C^*_{\max}$ for any instance of the $O2||C_{\max}$ problem coincides with the *standard lower bound*

$$\bar{C} = \max_{i,j}\left\{\sum_{i=1}^{m} p_{ji}, \sum_{j=1}^{n} p_{ji}\right\}. \tag{1}$$

Following [18] we refer to this property of an instance (and its optimal schedule) as *normality*.

The open shop with three machines is NP-hard [13], although the existence of a pseudopolynomial algorithm for $O3||C_{\max}$ is still an open question. A PTAS for this problem can be found in [23]. On the other hand, the $O||C_{\max}$ problem is strongly NP-hard even in the case when integer processing times do not exceed 2 [26]. Normality for the three-machine open shop cannot be guaranteed: there exists an instance of $O3||C_{\max}$ for which $C^*_{\max} = \frac{4}{3}\bar{C}$ [22]. However this is not the case for the open shop problem with *allowed preemption*. This problem is denoted as $Om|pmtn|C_{\max}$ and has the following distinction. It is allowed to interrupt the processing of any operation at any time moment and resume it later. That means that a feasible schedule may contain not a single processing interval for operation O_{ji}, but a series of *fragments* $[s_1(O_{ji}), c_1(O_{ji})], \ldots, [s_{\xi_{ji}}(O_{ji}), c_{\xi_{ji}}(O_{ji})]$. Total length of these partial intervals coincides with the processing time p_{ji}. The $O|pmtn|C_{\max}$ problem is polynomially solvable and the optimal makespan always coincides with the standard lower bound $\bar{C}$.

The notation *pmtn* for some scheduling problem means, that we are allowed to interrupt any operation processing as many times as we need, there is no restriction. On the other hand, the absence of this notations means a polar opposite case: no operation can be interrupted at all. One can imagine plenty of intermediate situations, which are not covered by these edge cases. However the investigation of scheduling problems with somehow restricted preemption is scarce. The only such case that comes to mind is the problem with limited number of interruptions (see *e.g.* [3]). The notation for this case is *k-pmtn* and it means, that only schedules with at most k interruptions are considered feasible. In this paper we introduce another type of restricted preemption, described by a set $\mathcal{A}$ of jobs which are *allowed to be interrupted*. We use notation $\mathcal{A}$-*pmtn* for this situation.

The routing open shop problem is introduced in [1, 2]. The input of this problem combines inputs from open shop and TSP in the following manner. An edge-weighted graph $G = \langle V, E\rangle$ (the input of the TSP) represents the transportation network, nodes of which contain immovable jobs from the input of the open shop problem. One of the nodes is a *depot*. Mobile machines are initially located at the depot, and have to process the jobs while traversing the transportation network. Machine has to be at the same location with a job it processes. Travel times between the nodes depend on the distance function $\mathrm{dist}(u, v)$, which satisfy the triangle inequality. After completing all operations, machines have to return to the depot. The goal is to minimize the completion time of the last machine's activity $R_{\max}$, *i.e.* performing an operation or returning back to the depot. We denote this problem as $ROm||R_{\max}$ or $ROm|G = X|R_{\max}$ in case we want to specify the structure X of the transportation network. Notation $\overrightarrow{R}Om||R_{\max}$ and $\overrightarrow{R}Om|G = X|R_{\max}$ is used for the problem with *asymmetric travel times*. In contrast to the classical open shop, the routing open shop problem is already NP-hard in the simplest case $RO2|G = K_2|R_{\max}$, *i.e.* the case with two machines and two nodes [2]. A description of FPTAS for this case can be found in [15]. Approximation algorithms for $RO2||R_{\max}$ and $ROm||R_{\max}$ with best known up to date worst-case performance guarantees are described in [6] and [16], respectively. The generalization of the problem with *unrelated travel times*, in which the distance between nodes is machine-specific, is investigated in [4, 7, 9, 14]. In this case an acronym *Rtt* (stands for "unRelated Travel Times") is added to the second field of the problem notation.

The preemptive version of the routing open shop problem is only superficially explored. It is known that the $RO2|pmtn, G = K_2|R_{\max}$ is polynomially solvable, while $RO|pmtn, G = K_2|R_{\max}$ is strongly NP-hard [21]. The complexity status of the problem with $m = \text{const}$ machines and $k = \text{const}$ nodes of transportation network is still unknown unless $m = k = 2$. A few polynomially solvable cases of the $RO2|pmth|R_{\max}$ are given in [5, 8].

This paper is focused on the research of the $\overrightarrow{R}O2|pmtn, Rtt, G = tree|R_{\max}$ problem, *i.e.* two-machine preemptive routing open shop with asymmetric and machine-dependent distances with the transportation network being a tree, massively generalizing the results from [5]. We describe the instance reduction of the problem under investigation to a simpler case of $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt, G = chain|R_{\max}$ preserving the standard lower bound and present new polynomially solvable cases, based on that transformation. The rest of the paper is organized as follows. Section 2 contains the detailed description of the problem under investigation. All necessary details concerning the instance transformation can be found in section 3. New results are given in section 4, followed by conclusion remarks and ideas for future research in section 5.

2. Preliminary notes

Let us give a formal description of the $\overrightarrow{R}O2|Rtt, \mathcal{A}\text{-}pmtn|R_{\max}$ problem, *i.e.* the asymmetric routing open shop problem with unrelated travel times and restricted preemption allowed.

An input of the problem consists of a connected graph $G = \langle V, E\rangle$ (transportation network), selected node $v_0 \in V$ (depot), set of machines $\mathcal{M} = \{M_1, M_2\}$, set of jobs $\mathcal{J} = \{J_1, \dots, J_n\}$, partitioned into non-empty subsets $\{\mathcal{J}(v)|v \in V\}$ with $\mathcal{J}(v)$ being the set of jobs, located at node v. A subset of interruptible jobs $\mathcal{A} \subseteq \mathcal{J}$ is given. For each machine M_i there is an asymmetric distance function dist_i on the set V^2. Its value $\mathrm{dist}_i(u, v)$ represents the travel time of machine M_i from u to v. Functions dist_i satisfy the triangle inequality. For each operation O_{ji} its processing time $p_{ji} \in \mathbb{Z}_{\geqslant 0}$ is given.

Note that standard problem with unrestricted preemption is a special case with $\mathcal{A} = \mathcal{J}$ and is denoted as $\overrightarrow{R}O2|Rtt, pmtn|R_{\max}$.

Machines are initially located at the depot and have to perform operations of each job while traveling over the transportation network: machine can only process operations of jobs from $\mathcal{J}(v)$ while being at the node v. Any number of machines can travel over the same edge in any direction at the same time. Different operations of the same job cannot be processed simultaneously, and each machine can process at most one operation at a time. Machines are allowed to interrupt the processing of any operation, but required to resume (and complete) the operation later. It means that any operation O_{ji} can be partitioned into any number of suboperations referred to as the *fragments* with total processing time equals to p_{ji}. Machines have to return to the depot after processing all the operations. We use notation $p_{ji}(I)$, $G(I)$, $\mathrm{dist}_i(I; u, v)$ and $\mathcal{J}(I; v)$, if we want to specify an input for a particular problem instance I.

A *schedule* S can be described by specifying the processing interval for all fragments of each operation:

$$S = \left\{ \left\{ [s_1(O_{ji}), c_1(O_{ji})], \dots, [s_{\xi_{ji}}(O_{ji}), c_{\xi_{ji}}(O_{ji})] \right\} \,\middle|\, i = 1, 2,\ j = 1, \dots, n \right\}.$$

Notation $s(O_{ji}) \doteq s_1(O_{ji})$ and $c(O_{ji}) \doteq c_{\xi_{ji}}(O_{ji})$ specifies the starting time and the completion time of the operation O_{ji} in some current schedule. We also use $s(O)$ and $c(O)$ to denote the starting and the completion times, respectively, for some fragment O. The following notation is used to indicate the total number of interruptions in schedule S:

$$\xi(S) \doteq \sum_{i,j} (\xi_{ji} - 1).$$

Note that $\xi(S) = 0$ means that schedule S is non-preemptive.

Definition 1. A schedule S for an instance I of the $\overrightarrow{R}O2|Rtt, \mathcal{A}\text{-}pmtn|R_{\max}$ problem is referred to as *feasible* if it satisfies the following conditions:

(1) For each operation O_{ji} inequalities $s_1(O_{ji}) \leqslant c_1(O_{ji}) \leqslant \cdots \leqslant s_{\xi_{ji}}(O_{ji}) \leqslant c_{\xi_{ji}}(O_{ji})$ hold and

$$\sum_{k=1}^{\xi_{ji}} \big(c_k(O_{ji}) - s_k(O_{ji})\big) = p_{ji}(I).$$

(2) Let fragments O and O' belong to the same job or are processed by the same machine. Then

$$\big(s(O), c(O)\big) \cap \big(s(O'), c(O')\big) = \emptyset.$$

(3) If operation of job $J_j \in \mathcal{J}(v)$ is the first to start by machine M_i then

$$s(O_{ji}) \geqslant \mathrm{dist}_i(I; v_0, v).$$

(4) If machine M_i processes a fragment O of operation O_{ji} before the processing of a fragment O' of operation $O_{j'i}$, $J_j \in \mathcal{J}(v)$, and $J_{j'} \in \mathcal{J}(v')$, then

$$s(O') \geqslant c(O) + \mathrm{dist}_i(I; v, v').$$

(5) For each $J_j \notin \mathcal{A}$ and each machine M_i

$$\xi_{ji} = 1.$$

Suppose an operation of job $J_j \in \mathcal{J}(v)$ is the last to be processed by machine M_i in a schedule S. Then we define the *return time* of machine M_i as

$$R_i(S) \doteq c(O_{ji}) + \mathrm{dist}_i(v, v_0).$$

The *makespan* of a schedule S is $R_{\max}(S) \doteq \max\limits_i R_i(S)$. The goal is to find a feasible schedule minimizing the makespan.

We use the following notation for some problem instance I.

- $\ell_i(I) \doteq \sum\limits_{j=1}^{n} p_{ji}(I)$ — the *load* of machine M_i;
- $\ell_{\max}(I) \doteq \max\limits_i \ell_i(I)$ — the maximal machine load;
- $d_j(I) \doteq \sum\limits_{i=1}^{2} p_{ji}(I)$ — the *duration* of job J_j;
- $d_{\max}(I; v) \doteq \max\limits_{J_j \in \mathcal{J}(v)} d_j(I)$ — the maximal job duration at node v;
- $\Delta(I; v) \doteq \sum\limits_{J_j \in \mathcal{J}(v)} d_j(I)$ — the *load* of node v;
- $\Delta(I) \doteq \sum\limits_{v \in V} \Delta(I; v)$ — the *total load* of instance I;
- $T_i^*(I)$ — the optimum for an underlying TSP for machine M_i, i.e. the length of the shortest cyclic route of machine M_i visiting each node at least once;
- $R^*_{\max}(I)$ — the optimal makespan;
- $\overleftrightarrow{\mathrm{dist}}_i(u, v) = \mathrm{dist}_i(u, v) + \mathrm{dist}_i(v, u)$;
- $\overleftrightarrow{\mathrm{dist}}_{\min}(u, v) = \min\{\mathrm{dist}_1(u, v) + \mathrm{dist}_2(v, u), \mathrm{dist}_2(u, v) + \mathrm{dist}_1(v, u)\}$.

We omit I from the notation in case when it does not lead to a confusion.

We use the following *standard lower bound* on the optimum for the routing open shop problem with asymmetric transportation network and unrelated travel times:

$$\bar{R}(I) \doteq \max\left\{ \max_i\big(\ell_i(I) + T_i^*(I)\big), \max_{v \in V}\big(d_{\max}(I; v) + \overleftrightarrow{\mathrm{dist}}_{\min}(I; v_0, v)\big)\right\}. \tag{2}$$

Note that special case of this bound for a problem with identical travel times was introduced in [1]. It also coincides with $\bar{C}$ in case all travel times are zero (in this case our problem is reduced to the classical open shop problem).

In the remainder of this paper we use simplified notation for the operations of each job J_j: a_j and b_j instead of O_{j1} and O_{j2}, respectively. Moreover, we use the same notation (a_j and b_j) for operations' processing times whenever it does not lead to a confusion.

3. Instance reduction

In this section we show, how the standard instance reduction technique can be used to transform any instance of $\overrightarrow{R}O2|pmtn, Rtt, G = tree|R_{\max}$ into a simplified instance of $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt, G = chain|R_{\max}$, preserving the standard lower bound. The important property of this procedure is the *reversibility*, which allows to treat any schedule for the transformed instance as a feasible schedule for the initial one. The procedure is based on two types of instance transformation: *job aggregation* (also known as *job grouping*) and *terminal edge contraction*, described in detail in [10] for the $RO2|G = tree|R_{\max}$ problem and in [7] for the $RO2|Rtt, G = tree|R_{\max}$ problem. Our case $\overrightarrow{R}O2|Rtt, G = tree|R_{\max}$ implies only subtle differences to these operations.

Definition 2. Let I be an instance of the problem $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt|R_{\max}$ with graph $G = \langle V; E\rangle$, and $\mathcal{K} \subseteq \mathcal{J}(I; v)$ for some $v \in V$. Then by *aggregation* of the set $\mathcal{K}$ we understand combining the jobs from $\mathcal{K}$ to treat the set as a single job with combined processing times:

$$\mathcal{J}(I'; v) \doteq \mathcal{J}(I; v) \setminus \mathcal{K} \cup \{J_{j_\mathcal{K}}\},\ \forall i = 1, \ldots, m\ p_{j_\mathcal{K} i}(I') \doteq \sum_{J_j \in \mathcal{K}} p_{ji}(I),$$
$$\forall u \neq v\ \mathcal{J}(I'; u) = \mathcal{J}(I; u).$$

(Here $j_\mathcal{K}$ is some new job index. A job $J_{j_\mathcal{K}}$ is to replace the set of jobs $\mathcal{K}$.)

As soon as aggregation leads to a new job $J_{j_\mathcal{K}}$ in $\mathcal{J}(I'; v)$, it is possible that $d_{j_\mathcal{K}} > d_{\max}(I; v)$. Specifically, (2) implies

$$\bar{R}(I') > \bar{R}(I) \text{ if and only if } d_{j_\mathcal{K}} > \bar{R}(I) - \overleftrightarrow{\operatorname{dist}}_{\min}(v_0, v). \tag{3}$$

We use job aggregation to simplify the instance preserving the standard lower bound. Such an aggregation is referred to as a *valid* one.

Definition 3. A node $v \in V$ of an instance I of the $ROm|\mathcal{A}\text{-}pmtn, Rtt|R_{\max}$ problem is referred to as *overloaded* if

$$\Delta(I; v) > \bar{R}(I) - \overleftrightarrow{\operatorname{dist}}_{\min}(I; v_0, v).$$

Otherwise the node is called *underloaded*.

The job aggregation of the set $\mathcal{J}(I; v)$ is valid if and only if the node v is underloaded. Therefore, any node containing single job is an underloaded one.

Now let us describe the terminal edge contraction operation.

Definition 4. Let $v \in V \setminus \{v_0\}$ be some terminal node in graph G, containing a single job J_j in an instance I of the $\overrightarrow{R}Om|\mathcal{A}\text{-}pmtn, Rtt|R_{\max}$ problem. Let $e = [v, u] \in E$ be the edge incident to v. By the *contraction* of the edge e we understand the following instance transformation:

$$\mathcal{J}(I'; u) \doteq \mathcal{J}(I; u) \cup \{J_{j'}\};\ p_{j'i}(I') \doteq p_{ji}(I) + \overleftrightarrow{\operatorname{dist}}_i(v, u);$$
$$G(I') \doteq G(I) \setminus \{v\};\ \mathcal{A}(I') \doteq \mathcal{A}(I) \setminus \{J_{j'}\}.$$

We want to perform an edge contraction operation only if it does not lead to the growth of the standard lower bound. Otherwise, the edge is called *overloaded*. The following definition describes the exact condition, under which an edge is overloaded.

Definition 5. Let $v \in V \setminus \{v_0\}$ be some terminal node in graph G, containing a single job J_j in instance I of the $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt|R_{\max}$ problem. Let $e = [v, u] \in E$ be the edge incident to v. The edge e is referred to as *overloaded* if

$$d_j(I) + \overleftrightarrow{dist}_1(I; v_0, v) + \overleftrightarrow{dist}_2(I; v_0, v) > \bar{R}(I) - \overleftrightarrow{dist}_{\min}(I; v_0, v). \tag{4}$$

For any problem instance I, we denote the number of overloaded nodes by $L_V(I)$ and the number of overloaded edges by $L_E(I)$.

Proposition 1. *Let I be an instance of the problem $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt|R_{\max}$. Then $L_V(I) + L_E(I) \leqslant 1$.*

The proof is very similar to that in [7, 10].

The following procedure generalizes the analogous one from [5] for the case with asymmetric distances and unrelated travel times.

The instance reduction procedure.

`INPUT:` An instance I of the problem $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt|R_{\max}$.

`OUTPUT:` A simplified instance $\tilde{I}$ of the problem $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt|R_{\max}$.

1. **For each** $v \in V$
   - 1.1. **If** v is underloaded **then** perform the job aggregation of $\mathcal{J}(v)$
   - 1.2. **Else** Perform the job aggregations at v until no valid job aggregation at v is possible.
2. **For each** *terminal node* $v \neq v_0$:
   - 2.1. $e \doteq [u, v]$ (e is incident to v, u is adjacent to v),
   - 2.2. **If** e *is underloaded*
     - 2.2.1. Let J_j is the only job in $\mathcal{J}(v)$,
     - 2.2.2. Perform the contraction of e: $J_j \to J_{j'}$, remove $J_{j'}$ from the set of $\mathcal{A}$.
     - 2.2.3. **If** u is underloaded **then** perform the job aggregation of $\mathcal{J}(u)$ and remove the resulting job from set $\mathcal{A}$.
3. **End.**

This procedure can be implemented to run in $O(n)$ time. It obtains so-called *irreducible instance*, *i.e.* an instance for which no valid job aggregation is possible.

The following Lemma describes all possible variants of the reduced instance for the problem $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt, G = tree|R_{\max}$.

Lemma 1. *Let I be an instance of $\overrightarrow{R}O2|\mathcal{A}\text{-}pmtn, Rtt, G = tree|R_{\max}$ and $\tilde{I}$ is obtained from I using the instance reduction procedure. Then $\bar{R}(\tilde{I}) = \bar{R}(I)$ and the graph $G(\tilde{I})$ satisfies exactly one of the following conditions:*

1. *$G(\tilde{I})$ has a single node v_0;*
2. *$G(\tilde{I})$ is a chain connecting v_0 with a node v with single job at each node, and the edge incident to v is overloaded;*
3. *$G(\tilde{I})$ is a chain connecting v_0 with an overloaded node v, and each node except v contains only one job.*

The proof is similar to that in [7]. Moreover, it is proved in [7] for symmetric distances, that in cases 1 and 2 of Lemma 1 there exists a non-preemptive schedule for the simplified instance with makespan of $\bar{R}$. The prove can be easily reformulated to fit our asymmetric problem.

The third case (with an overloaded node) is partly covered in the next section.

4. New polynomially solvable cases

The following Lemma covers a special simplified instance.

Lemma 2. *Let I be an irreducible instance of $\overrightarrow{R}O2|Rtt, \mathcal{A}\text{-}pmtn, G = chain|R_{\max}$, with $G(I)$ being a chain connecting v_0 and an overloaded node $v_k = v$, $k \geqslant 1$, each underloaded node v_p contains a single job J_p, $0 \leqslant p \leqslant k-1$ and $\mathcal{J}(v_k)$ contains at most three jobs J_α, J_β and J_k (job J_k might be a dummy one), and $J_\beta \in \mathcal{A}$. Then a feasible schedule S for I with $\xi(S) \leqslant 2$ and $R_{\max}(S) = \bar{R}(I)$ can be built in linear time.*

Proof. Note that by irreducibility of instance I and (3) we have

$$d_\alpha + d_\beta > \bar{R}(I) - \overleftrightarrow{\mathrm{dist}}_{\min}(v_0, v_k). \tag{5}$$

Denote $\overrightarrow{T_i} \doteq \mathrm{dist}_i(v_0, v_k)$ and $\overleftarrow{T_i} \doteq \mathrm{dist}_i(v_k, v_0)$, $i = \{1, 2\}$.

Without loss of generality we may assume

$$\overrightarrow{T_1} + \overleftarrow{T_2} \geqslant \overrightarrow{T_2} + \overleftarrow{T_1} \tag{6}$$

(this can be achieved by renumeration of machines).

Then, by (5) and (6), we have

$$d_\alpha + d_\beta > \bar{R}(I) - (\overrightarrow{T_2} + \overleftarrow{T_1}). \tag{7}$$

We use the following notation:

$$A = \sum_{j=0}^{k} a_j + \overrightarrow{T_1} = l_1 - (a_\alpha + a_\beta) + \overrightarrow{T_1}, \tag{8}$$

$$B = \sum_{j=0}^{k} b_j + \overleftarrow{T_2} = l_2 - (b_\alpha + b_\beta) + \overleftarrow{T_2}, \tag{9}$$

$$C = \bar{R}(I) - A - B. \tag{10}$$

Note that (2) implies

$$\sum_j d_j + \overrightarrow{T_1} + \overleftarrow{T_1} + \overrightarrow{T_2} + \overleftarrow{T_2} = l_1 + l_2 + \overrightarrow{T_1} + \overleftarrow{T_1} + \overrightarrow{T_2} + \overleftarrow{T_2} \leqslant 2\bar{R}. \tag{11}$$

By (7), (8), (9) and (11) we obtain

$$A + B = \sum_{j \neq \alpha, \beta} d_j + \overrightarrow{T_1} + \overleftarrow{T_2} < \bar{R}(I),$$

therefore C is positive.

Consider the following cases.

Case 1:

$$\sum_{j \neq \beta} d_j + \overrightarrow{T_1} + \overleftarrow{T_2} \leqslant \bar{R}. \tag{12}$$

Construct a non-preemptive schedule according to the partial order of operations shown in Fig. 1.

It is easy to observe using (2) and (12), that makespan of this schedule does not exceed $\bar{R}(I)$.

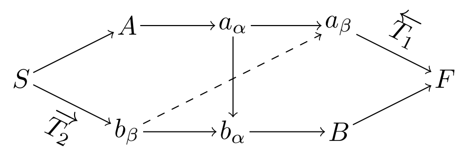

FIG. 1. Schedule diagram for Case 1.

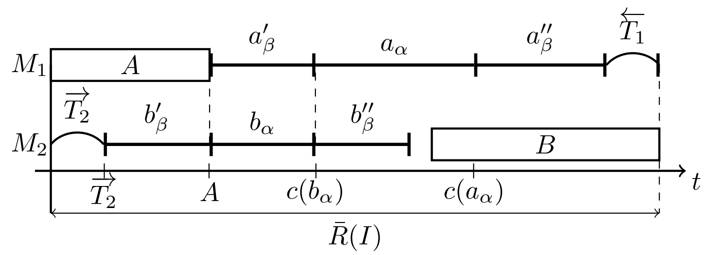

FIG. 2. Schedule for Case 2.1.1.

Case 2:

$$\sum_{j\neq\beta} d_j + \overrightarrow{T_1} + \overleftarrow{T_2} > \bar{R}. \quad (13)$$

Case 2.1:

$$d_\alpha \leqslant C + B - \overleftarrow{T_1}. \quad (14)$$

Case 2.1.1: $b_\alpha \leqslant C$.

Partition each of the operations of job J_α into at most two fragments in the following way:

$$a'_\beta = \min\{a_\beta, b_\alpha\},\ a''_\beta = a_\beta - a'_\beta,$$

$$b'_\beta = \min\{b_\beta, A - \overrightarrow{T_2}\},\ b''_\beta = b_\beta - b'_\beta.$$

Next build a feasible schedule shown in Fig. 2. Block B starts executing on the machine M_2 at time $\bar{R}(I) - B$.

The final schedule may differ from the presented one, due to the ambiguous definition of fragments of operations job J_β. However, due to the construction, the schedule has in any case the following properties:

$$\begin{aligned} &s(b_\alpha) = c(A) = A,\ s(a_\alpha) = c(b_\alpha),\\ &s(b'_\beta) = \overrightarrow{T_2},\ c(b'_\beta) \leqslant A,\\ &s(a'_\beta) = A,\ c(a'_\beta) \leqslant c(b_\alpha). \end{aligned} \quad (15)$$

By virtue of (13) and the construction of the schedule, $c(a_\alpha) > s(B)$. This inequality and conditions (15) imply the feasibility of the constructed schedule. Its makespan coincides with $\bar{R}$ by construction.

Case 2.1.2:

$$b_\alpha > C. \quad (16)$$

In this case, subtracting (16) from (14) results in

$$a_\alpha < B - \overleftarrow{T_1}. \tag{17}$$

Partition operation a_β as follows

$$a'_\beta = \min\{a_\beta, C\},\ a''_\beta = a_\beta - a'_\beta$$

and build a schedule as shown in Fig. 3.

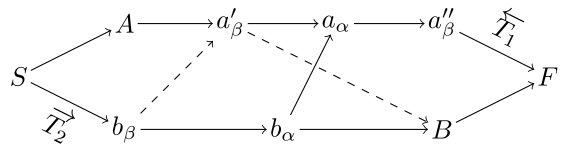

FIG. 3. Schedule diagram for Case 2.1.2.

By (2), (10) and (16) we have

$$b_\beta < A - \overrightarrow{T_2}. \tag{18}$$

Suppose the makespan of the schedule built exceeds $\bar{R}(I)$. Then given $a'_\beta \leqslant C$ and (18), there is the only one possibility for the critical path in the diagram from Fig. 3:

$$S \xrightarrow{\overrightarrow{T_2}} b_\beta \longrightarrow b_\alpha \longrightarrow a_\alpha \longrightarrow a''_\beta \xrightarrow{\overleftarrow{T_1}} F$$

Let us prove that its length does not exceed $\bar{R}(I)$ and the schedule is normal.

1. if $a_\beta > C$, therefore $a'_\beta = C$, then, given $\overrightarrow{T_2} + b_\alpha + b_\beta \leqslant A + C$, we get
$$R = \overrightarrow{T_2} + b_\beta + d_\alpha + a''_\beta + \overleftarrow{T_1} \leqslant A + a_\alpha + a_\beta + \overleftarrow{T_1} \leqslant \overline{R}(I);$$
2. if $a_\beta \leqslant C$, then $a''_\beta = 0$, and given (17), we have
$$R = \overrightarrow{T_2} + b_\beta + d_\alpha + \overleftarrow{T_1} \leqslant \overrightarrow{T_2} + b_\beta + b_\alpha + B - \overleftarrow{T_1} + \overleftarrow{T_1} \leqslant \overline{R}(I).$$

Case 2.2:

$$d_\alpha > C + B - \overleftarrow{T_1}. \tag{19}$$

Case 2.2.1:

$$a_\alpha < B - \overleftarrow{T_1}. \tag{20}$$

In this case, subtracting (19) from (20) results in (16), and we can build a normal schedule exactly as in the Case 2.1.2.

Case 2.2.2: $a_\alpha \geqslant B - \overleftarrow{T_1}$.

In this case partition operation b_β as follows

$$b'_\beta = \min\{b_\beta,\ \overline{R}(I) - d_\alpha - \overrightarrow{T_2} - \overleftarrow{T_1}\},\quad b''_\beta = b_\beta - b'_\beta \tag{21}$$

and build a schedule as shown in Fig. 4. It follows from (21) that $b'_\beta + d_\alpha + \overrightarrow{T_2} + \overleftarrow{T_1} \leqslant \overline{R}(I)$. Suppose the makespan of the schedule built exceeds $\bar{R}(I)$. Then given $a'_\beta \leqslant C$ and (18), there is the only one possibility for the critical path in the diagram from Fig. 4:

$$S \longrightarrow A \longrightarrow a_\beta \longrightarrow b''_\beta \longrightarrow B \longrightarrow F$$

Estimate its length:

1. if $b_\beta \leqslant \bar{R} - d_\alpha - \overrightarrow{T_2} - \overleftarrow{T_1}$, *i.e.* $b''_\beta = 0$, then

$$R = A + a_\beta + b''_\beta + B = A + a_\beta + B \leqslant A + a_\beta + a_\alpha + \overleftarrow{T_1} \leqslant \bar{R}.$$

2. if $b_\beta > \bar{R} - d_\alpha - \overrightarrow{T_2} - \overleftarrow{T_1}$, *i.e.* $b''_\beta = b_\beta - \bar{R} + d_\alpha + \overrightarrow{T_2} + \overleftarrow{T_1}$, then, given $A + a_\beta \leqslant \bar{R} - \overleftarrow{T_1} - a_\alpha$, we get

$$R = A + a_\beta + b''_\beta + B \leqslant \bar{R} - \overleftarrow{T_1} - a_\alpha + b_\beta - \bar{R} + d_\alpha + \overrightarrow{T_2} + \overleftarrow{T_1} + B = \\ = \overrightarrow{T_2} + b_\beta + b_\alpha + B \leqslant \bar{R}.$$

Thus, we have considered all possible cases and have proved the existence of a valid schedule of length $\bar{R}(I)$ for each of them.

Note that the complexity of constructing an optimal schedule is linear from the number of operations of instance I, which in our case is $O(k)$. The theorem is proved. □

Now we can present the main result of this paper.

Theorem 1. *Let I be an instance of $\overrightarrow{R}O2|Rtt, pmtn, G = tree|R_{\max}$ where all nodes of G, with possible exception of the depot, have degree of at most 2. Then a feasible schedule S for I with $\xi(S) \leqslant 2$ and $R_{\max}(S) = \bar{R}(I)$ can be built in linear time.*

Proof. In the case the initial instance contains an overloaded node, according to [5], there is a feasible schedule of makespan $\bar{R}(I)$ with at most one interruption, constructable in linear time.

If instance I contains only underloaded nodes, we apply the Instance reduction procedure. As a result, we obtain an irreducible instance $\tilde{I}$.

By Lemma 1 and remarks thereafter, we only need to consider the case when $G(\tilde{I})$ is a chain connecting v_0 to an overloaded node v.

Since in instance I all nodes are underloaded, after the first step of the instance reduction each node will contain single job. In addition, during the procedure of instance reduction, an overloaded node participates in only one edge contraction operation, since its original degree is two (not that it can not be terminal, as all the initial nodes are underloaded). Therefore, after applying the edge contraction operation, the overloaded node will contain at least two jobs, with one of them belonging to set $\mathcal{A}$. Thus we can use Lemma 2 to construct the desired schedule, which can be treated as a feasible schedule for the initial instance of the same makespan. The theorem is proved. □

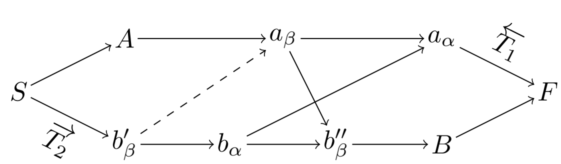

FIG. 4. Schedule diagram for Case 2.2.2.

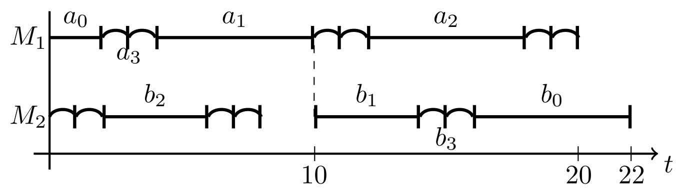

FIG. 5. An optimal schedule for instance $\tilde{I}$.

Remark 1. There are two types of tree structures obeying the conditions of Theorem 1: chains and spiders, providing that in case of a spider the node with degree greater than two is the depot.

The following example shows that the location of the depot in a spider transportation network is crucial for the existence of a normal schedule, even for a special case of symmetric identical travel times.

Example 1. Consider the following instance $\tilde{I}$ of the $RO2|pmtn, G = spider$ problem. Let $G(\tilde{I})$ is the complete bipartite graph $K_{1,3}$ (or *star* S_3), depot v_0 have degree 1 and node v_1 has degree 3. Each node $v_i, i = 0, \ldots, 3$ contains single job J_i with the following matrix of processing times $\left(\begin{smallmatrix} 2 & 0 & 6 & 6 \\ 6 & 0 & 4 & 4 \end{smallmatrix}\right)$. All travel times are unit. The standard lower bound for this instance is $\bar{R}(\tilde{I}) = 20$, and the optimal schedule is shown in Fig. 5; it has a makespan of 22. The optimality of this schedule can be shown as follows. In any feasible schedule with minimal routing (when each machine visits each node exactly once) preemption is redundant, and non-optimal routing would increase the makespan to at least 24. Now, as soon as we have only three jobs, it is easy to construct a few possible non-preemptive schedules to make sure that optimal makespan is indeed 22. Therefore $R^*(\tilde{I}) = \frac{11}{10}\bar{R}(\tilde{I})$.

5. Conclusion

The main result of this paper is a new polynomially solvable case for a generalized version of the two-machine routing open shop on a tree with preemption allowed. It follows from Theorem 1 that for any instance I of $\overrightarrow{R}O2|pmtn, Rtt, G = tree|R_{\max}$ with the property that the degree of each node $v \in V \setminus \{v_0\}$ is at most two, the optimum coincides with the standart lower bound $\bar{R}(I)$ and an optimal schedule can be constructed in $O(n)$ time, where n is the number of jobs. On the other hand, we find the research of scheduling problems with restricted preemption practically important and actual. Notation $\mathcal{A}$-*pmtn* generalizes both preemptive and non-preemptive scheduling problems. For example, it would be of interest to investigate a complexity status of some $\mathcal{A}$-*pmtn* problem, whose both preemptive and non-preemptive counterparts are polynomially solvable, but the preemption is not redundant and may influence the optimum. An example of such a problem is a *mixed shop* with at most two unit operations per job, which can be denoted as $X|p_{ji} = 1, \nu \leq 2|C_{\max}$. This problem is polynomially solvable [17], as well as its preemptive counterpart $X|pmtn, p_{ji} = 1, \nu \leq 2|C_{\max}$ [12]. It is shown in [12] that for $Xm|pmtn, p_{ji} = 1, \nu \leq 2|C_{\max}$ with $m \geqslant 4$ preemption is not redundant. Thus, we suggest the following

Question 1. Is $X4|\mathcal{A}\text{-}pmtn, p_{ji} = 1, \nu \leq 2|C_{\max}$ polynomially solvable?

Example 1 shows that the normality cannot be guaranteed for a general case of the $\overrightarrow{R}O2|pmtn, Rtt, G = tree|R_{\max}$ and even the $RO2|pmtn, G = tree|R_{\max}$ problem. This disproves the hypothesis from [5] about normality of any instance of $RO2|pmtn, G = tree|R_{\max}$ and implies the following

Question 2. Is there an instance I of the $RO2|pmtn, G = tree|R_{\max}$ problem such that $R^*_{\max}(I) > \frac{11}{10}\bar{R}(I)$?

As we discussed in the introduction section, the complexity of the routing open shop with preemption allowed is understudied. Therefore we find the following question to be the most intriguing:

Question 3. What are the complexity statuses of the following problems:

- $RO2|pmtn, G = tree|R_{\max}$;
- $RO2|pmtn, G = K_3|R_{\max}$;
- $RO3|pmtn, G = K_2|R_{\max}$;
- $ROm|pmtn, G = K_p|R_{\max}$ in case m and p are constant and $\max\{m, p\} > 2$.

References

[1] I. Averbakh, O. Berman, I. Chernykh, *A 6/5-approximation algorithm for the two-machine routing open-shop problem on a two-node network,* Eur. J. Oper. Res., **166**:1 (2005), 3–24. Zbl 1069.90037

[2] I. Averbakh, O. Berman, I. Chernykh, *The routing open-shop problem on a network: complexity and approximation,* Eur. J. Oper. Res., **173**:2 (2006), 531–539. Zbl 1109.90038

[3] O. Braun, G. Schmidt, *Parallel processor scheduling with limited number of preemptions,* SIAM J. Comput., **32**:3 (2003), 671–680. Zbl 1023.90021

[4] I. Chernykh, *Routing open shop with unrelated travel times,* in Kochetov, Yury (ed.) et al., *Discrete optimization and operations research. 9th international conference, DOOR 2016, Vladivostok, Russia, September 19–23, 2016. Proceedings*, Lecture Notes in Computer Science, **9869**, 2016, 272–283. Zbl 1385.90007

[5] I. Chernykh, *Sufficient conditions of polynomial solvability of the two-machine preemptive routing open shop on a tree,* in Evtushenko, Yury (ed.) et al., *Optimization and applications. 9th international conference, OPTIMA 2018, Petrovac, Montenegro, October 1–5, 2018. Revised selected papers*, Comput. Inf. Sci., **974**, 2019, 97–110. Zbl 1464.90026

[6] I. Chernykh, A. Kononov, S. Sevastyanov, *Efficient approximation algorithms for the routing open shop problem,* Comput. Oper. Res., **40**:3 (2013), 841–847. Zbl 1349.90330

[7] I. Chernykh, O. Krivonogova, *Efficient algorithms for the routing open shop with unrelated travel times on cacti,* in Jaćimović, Milojica (ed.) et al., *Optimization and applications. 10th international conference, OPTIMA 2019, Petrovac, Montenegro, September 30 – October 4, 2019. Revised selected papers*, Commun. Comput. Inf. Sci., **1145**, Springer, Cham, 2020, 1–15. Zbl 1473.90013

[8] I. Chernykh, M. Kuzevanov, *Sufficient condition of polynomial solvability of two-machine routing open shop with preemption allowed,* Intellektual'nye Sistemy, **17** (2013), 552–556.

[9] I. Chernykh, E. Lgotina, *How the difference in travel times affects the optima localization for the routing open shop,* in Khachay, Michael (ed.) et al., *Mathematical optimization theory and operations research. 18th international conference, MOTOR 2019, Ekaterinburg, Russia, July 8–12, 2019*, Lect. Notes Comput. Sci., **11548**, Springer, Cham, 2019, 187–201. Zbl 1439.90032

[10] I. Chernykh, E. Lgotina, *Two-machine routing open shop on a tree: instance reduction and efficiently solvable subclass,* Optimization Methods And Software, (2020). doi: 10.1080/10556788.2020.1734802

[11] N. Christofides, *Worst-case analysis of a new heuristic for the travelling salesman problem,* Technical Report, **388**, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburg, PA, 1976

[12] A. Dugarzhapov, A. Kononov, *A polynomial-time algorithm for the preemptive mixed-shop problem with two unit operations per job,* J. Sched., **19**:1 (2016), 61–72. Zbl 1341.90043

[13] T. Gonzalez, S. Sahni, *Open shop scheduling to minimize finish time,* J. Assoc. Comput. Mach., **23** (1976), 665–679. Zbl 0343.68031

[14] A. Khramova, I. Chernykh, *A new algorithm for the two-machine open shop and the polynomial solvability of a scheduling problem with routing,* J. Sched., **24**:4 (2021), 405–412. Zbl 1479.90098

[15] A. Kononov, *On the routing open shop problem with two machines on a two-vertex network,* J. Appl. Ind. Math., **6**:3 (2012), 318–331. Zbl 1324.90059

[16] A. Kononov, *$O(\log m)$-approximation for the routing open shop problem,* RAIRO, Oper. Res., **49**:2 (2015), 383–391. Zbl 1310.90014

[17] A. Kononov, S. Sevastyanov, M. Sviridenko, *A Complete 4-parametric complexity classification of short shop scheduling problems,* J. Sched., **15**:4 (2012), 427–446. Zbl 1280.90057

[18] A. Kononov, S. Sevastianov, I. Tchernykh, *When difference in machine loads leads to efficient scheduling in open shops,* Ann. Oper. Res., **92** (1999), 211–239. Zbl 0958.90040

[19] E. Lawler, J. Lenstra, A. Rinnooy Kan, D. Shmoys, *Sequencing and scheduling: Algorithms and complexity,* in *Logistics Of Production And Inventory*, Handbooks in Operations Research and Management Science, **4**, Elsevier, 1993, 445–522.

[20] M. Pinedo, L. Schrage, *Stochastic shop scheduling: A survey,* Deterministic and stochastic scheduling, Proc. NATO Adv. Study Res. Inst., Durham/Engl. 1981, (1982), 181–196. Zbl 0498.68022

[21] A. Pyatkin, I. Chernykh, *The open shop problem with routing at a two-node network and allowed preemption,* J. Appl. Ind. Math. , **6**:3 (2012), 346–354. Zbl 1324.90020

[22] S. Sevastyanov, I. Tchernykh, *Computer-aided way to prove theorems in scheduling,* in Bilardi, Gianfranco (ed.) et al., *Algorithms - ESA '98. 6th annual European symposium, Venice, Italy, August 24–26, 1998. Proceedings*, Lect. Notes Comput. Sci., **1461**, Springer, Berlin, 1998, 502–513. Zbl 0929.90040

[23] S. Sevastianov, G. Woeginger, *Makespan minimization in open shops: A polynomial time approximation scheme,* Math. Program., **82**:1-2 (1998), 191–198. Zbl 0920.90075

[24] A. Soper, *A cyclical search for the two machine flow shop and open shop to minimise finishing time,* J. Sched., **18**:3 (2015), 311–314. Zbl 1328.90060

[25] D. de Werra, *Graph-theoretical models for preemptive scheduling,* Advances In Project Scheduling, (1989), 171–185.

[26] D. Williamson, L. Hall, J. Hoogeveen, C. Hurkens, J. Lenstra, S. Sevast'janov, D. Shmoys, *Short shop schedules,* Oper. Res., **45**:2 (1997), 288–294. Zbl 0890.90112

Polina Matveevna Agzyamova
Novosibirsk State University,
1, Pirogova str.,
Novosibirsk, 630090, Russia
Email address: p.agzyamova@g.nsu.ru

Ilya Dmitrievich Chernykh
Sobolev Institute of Mathematics,
4, Koptyuga ave.,
Novosibirsk, 630090, Russia
Email address: idchern@math.nsc.ru