International Journal of Production Research

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tprs20

# Integrated job-shop scheduling in an FMS with heterogeneous transporters: MILP formulation, constraint programming, and branch-and-bound

**Amir Ahmadi-Javid, Maryam Haghi & Pedram Hooshangi-Tabrizi**

**To cite this article:** Amir Ahmadi-Javid, Maryam Haghi & Pedram Hooshangi-Tabrizi (2024) Integrated job-shop scheduling in an FMS with heterogeneous transporters: MILP formulation, constraint programming, and branch-and-bound, International Journal of Production Research, 62:9, 3288-3304, DOI: 10.1080/00207543.2023.2230489

**To link to this article:** https://doi.org/10.1080/00207543.2023.2230489

Published online: 11 Aug 2023.

INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH
2024, VOL. 62, NO. 9, 3288–3304
https://doi.org/10.1080/00207543.2023.2230489



# Integrated job-shop scheduling in an FMS with heterogeneous transporters: MILP formulation, constraint programming, and branch-and-bound

Amir Ahmadi-Javid [a], Maryam Haghi [b] and Pedram Hooshangi-Tabrizi [c]

[a]Department of Industrial Engineering & Management Systems, Amirkabir University of Technology, Tehran, Iran; [b]Department of Decision Sciences, HEC Montréal, Montréal, Canada; [c]Department of Mechanical, Industrial & Aerospace Engineering, Concordia University, Montréal, Canada

**ABSTRACT**

Current studies on scheduling of machines and transporters assume that either a single transporter or an infinite number of homogeneous transporters such as AGVs or mobile robots are available to transport semi-finished jobs, which seems very restrictive in practice. This paper addresses this gap by studying a job-shop scheduling problem that incorporates a limited number of heterogeneous transporters, where the objective is to minimize the makespan. The problem is modelled using mixed-integer linear programming and constraint programming. Different structure-based branch-and-bound algorithms with two lower-bounding strategies are also developed. A comprehensive numerical study evaluates the proposed models and algorithms. The research demonstrates that the adjustment of the proposed MILP model outperforms the existing formulation when applied to the homogeneous case. The study also uncovers interesting practical implications, including the analysis of the impact of different transporter types in the system. It shows that utilizing a fleet of heterogeneous transporters can improve the overall performance of the job shop compared to a relevant homogeneous case. The importance of the study is emphasized by highlighting the negative consequences of disregarding transporters' differences during the scheduling phase.

**ARTICLE HISTORY**
Received 10 February 2020
Accepted 12 May 2023

**KEYWORDS**
Job-shop scheduling; flexible manufacturing system (FMS); mobile robots and automated guided vehicles (AGVs); minimal time lags; mixed-integer linear programming (MILP); constraint programming (CP); structure-based branch-and-bound (B&B)

## 1. Introduction

In a production environment, resources can be machines, material handling equipment, or employees. Production and scheduling plans involve allocating resources to the activities that need to be executed (Pinedo 2005). Basic scheduling problems often neglect the transportation times of carrying semi-finished jobs between machines, while in practice different types of transporters are usually required for transportation. Therefore, it seems unjustified to assume that the time required to transfer a job between machines is always negligible. Although material-handling constraints make scheduling problems more complicated, the coordination of resources such as machines and transporters can definitely increase the efficiency of a manufacturing system.

Machine and transporter scheduling problems are highly interconnected in a production system, as they both involve the allocation and coordination of resources to ensure efficient operations. Many papers have studied integrated scheduling of processing operations on machines and transportation tasks on material handling equipment. As an earlier pertinent work, Bilge and Ulusoy (1995) studied a basic version of such a problem in a job-shop environment. They presented an optimisation formulation and a heuristic to solve the problem. Xie and Allen (2015) provided a comprehensive review of job-shop scheduling problems considering material handling systems. A survey by Nouri, Driss, and Ghédira (2016a) classified papers on job-shop scheduling problems that consider transportation resources such as Automated Guided Vehicles (AGVs) and robots.

From a computational perspective, incorporating transporter scheduling into the shop scheduling increases the complexity. For example, the two-machine flow-shop and open-shops with makespan criterion can be solved in polynomial time ($O(nlogn)$ and $O(n)$ for $n$ jobs), whereas their integrations with transportation time lags (unlimited transporters) are strongly NP-hard (Dell'Amico 1996; Hurink and Knust 2001; Yu, Hoogeveen, and Lenstra 2004).

Traditionally, two types of transportation systems have been considered in integrated shop scheduling problems, where either a *single* transporter or an *unlimited* number of transporters are used to perform transportation tasks. In many practical settings, the number of transporters is often limited, and they are not necessarily similar

**CONTACT** Amir Ahmadi-Javid ahmadi_javid@aut.ac.ir Department of Industrial Engineering & Management Systems, Amirkabir University of Technology, Tehran, Iran

(homogeneous) in terms of technology, speed, capacity, and other characteristics. Furthermore, transporters can be categorized as either *allocated* or *free*. An allocated transporter can only move in a predetermined area, like between specific machines, while a free transporter is able to move between any pairs of machines.

Table 1 summarises the most relevant studies on simultaneously scheduling of machines and material handling systems in classic shop environments (i.e. flow shops, job shops, and open shops). Note that the interaction of scheduling and transportation is also of interest in more complex environments. For example, Ahmadi-Javid and Hooshangi-Tabrizi (2017) integrated employee time tabling into the scheduling of machines and transporters. Lin, Chiu, and Chang (2019) presented a simulation-based optimisation algorithm for an integrated scheduling problem in an uncertain environment. Dang, Nguyen, and Rudová (2019) studied a problem in which mobile robots can do value-added tasks using their manipulation arms without any human intervention. Sun et al. (2021) studied robotic job-shop scheduling with deadlock and robot movement. Homayouni and Fontes (2021) studied a flexible job shop scheduling problem in which jobs needed to be moved around the shop-floor by a set of transporters. Li et al. (2021) solved the flexible job shop scheduling problem with crane transportation processes considering energy consumptions. Ren et al. (2021) proposed a proactive-reactive method for adapting to the dynamic changes in a dynamic flexible job-shop scheduling problem with constraints of transportation resources and transportation time. Li et al. (2022) studied a real-time data-driven dynamic flexible job shop scheduling problem with insufficient transportation resources. Cai et al. (2022) developed a real-time scheduling model and algorithm for a dynamic job shop where there was a new kind of release moment of task information that could give AGVs the longest time to prepare for the task. Schulz, Schönheit, and Neufeld (2022) developed a bi-objective carbon-efficient distributed permutation flow shop with transportation decisions.

As can be seen in Table 1, the existing papers on the integration of shop scheduling and transportation assume that all the transporters are homogeneous. From Table 1, one can conclude that the problem of the integrated scheduling of machines and heterogeneous transporters has not yet studied and solved by an effective exact solution method. To address this gap, this paper considers the integrated scheduling of machine and heterogeneous transporters in a job-shop environment.

Flexible Manufacturing Systems (FMSs) can serve as a motivating context for the problem presented in this paper (ElMaraghy 2005; Yadav and Jayswal 2018). An FMS is an integrated system of flexible manufacturing machines and a material handling system where all the system components can be controlled (Buzacott and Yao 1986); consequently, scheduling in such an environment is a challenging task (Hu et al. 2020). In FMSs, depending on the type of products to be handled, manufacturing infrastructure, and economical considerations, various solutions for material handling can operate continuously or on demand. Among these solutions, one can mention belt, roller and vertical conveyors, elevators, AGVs, and fixed and mobile robots (Martinez-Barbera and Herrero-Perez 2010). In FMSs, transports are basically different. Even for one class of transporters such as AGVs or mobile robots, one cannot assume that they are homogenous due to technological evolution. Heterogeneous transportation fleets have been broadly considered in other areas of transportation literature such as vehicle routing (Baldacci, Battarra, and Vigo 2008; Koç et al. 2016). Indeed, heterogeneity increases the complexity of planning as the task allocation becomes important, which is a challenge in planning multi-robot systems (Parker, Rus, and Sukhatme 2016; Verma and Ranga 2021).

In the integrated scheduling problem studied in this paper, machines and heterogeneous transporters are simultaneously scheduled in a job-shop environment. It is also assumed that the transporters are limited and free. The aim is to find a scheduling of the machines and transporters with the minimum makespan, which is the maximum completion time of all jobs. According to the $\alpha|\beta|\gamma$ notation of Graham et al. (1979), the studied problem can be called as $J|T(free, lim, hetero)|C_{max}$, following Ahmadi-Javid and Hooshangi-Tabrizi (2015). In this notation, $J$ represents a job-shop environment, $T$ is related to the transportation setting, which indicates *free*, limited (*lim*), and heterogeneous (*hetero*) transporters. Finally, $C_{max}$ or makespan is the objective function. The results of our paper can be used in less complex scheduling environments such as flow shops because similar problems have not been studied for them (Rossit, Tohmé, and Frutos 2018). To solve the proposed problem $J|T(free, lim, hetero)|C_{max}$, it is cast as Mixed-Integer Linear Programming (MILP) and Constraint Programming (CP) models. Also, different Structured-Based Branch-and-Bound (SB-B&B) algorithms with two lower bounding procedures are developed. The solution methods (and the adjustment of the best heuristic for the homogenous case) are compared on a set of adjusted benchmark instances.

The contributions of this paper can be summarised as follows:

- For the first time, the integrated scheduling of machines and *heterogeneous* transporters is studied in the literature of shop scheduling.

**Table 1.** An overview of the studies on integrated shop scheduling with transporters.

| | Scheduling problem | | | | | Transporters assumptions: Movement | | Transporters assumptions: Multiplicity | | | Transporters assumptions: Type | | Solution methodology | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Flow-shop | | | | | | | | | | | | | | | |
| Authors Year | General | Permutation | Flexible | Job-shop | Open-shop | Allocated | Free | Single | Unlimited | Limited | Homogeneous | Heterogeneous | Modelling | Heuristic | Meta-Heuristic | Description |
| Raman (1986) | ✓ | | | | | | ✓ | | | ✓ | ✓ | | ✓ | ✓ | | IP-Return to L/U repeatedly |
| Langston (1987) | | | ✓ | | | | | ✓ | | | | | | ✓ | | Heuristics |
| Bilge and Ulusoy (1995) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | ✓ | | MINLP, STW |
| Ulusoy, Sivrikaya-ŞerifoÇğlu, and Bilge (1997) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | GA |
| Anwar and Nagi (1998) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | ✓ | | MIP |
| Sabuncuoglu and Karabuk (1998) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | FBS |
| Rebaine and Strusevich (1999) | | | | | ✓ | | | | ✓ | | ✓ | | | ✓ | | Two machines |
| Hurink and Knust (2002) | | | | ✓ | | | | ✓ | | | | | | | ✓ | TS |
| Abdelmaguid et al. (2004) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | HGA |
| Hurink and Knust (2005) | | | | ✓ | | | | ✓ | | | | | | | ✓ | TS |
| Lacomme, Moukrim, and Tchernev (2005) | | | | ✓ | | | | ✓ | | | | | | ✓ | | HBB coupled with DES |
| Khayat, Langevin, and Riopel (2006) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | | | MILP, CP |
| Reddy and Rao (2006) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | HGA |
| Jerald et al. (2006) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | GA |
| Caumond et al. (2009) | | | | ✓ | | | | ✓ | | | | | ✓ | | | MILP |
| Deroussi, Gourgand, and Tchernev (2008) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | Integrated SA, LS |
| Naderi, Ahmadi-Javid, and Jolai (2010) | | ✓ | | | | ✓ | | ✓ | ✓ | | | | ✓ | ✓ | ✓ | MIP, AI |
| Chaudhry, Mahmood, and Shami (2011) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | GA |
| Kumar, Janardhana, and Rao (2011) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | ✓ | | Heuristics |
| Lacomme, Larabi, and Tchernev (2013) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | ✓ | | LS |
| Zhang, Manier, and Manier (2014) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | ✓ | | Shifting bottleneck heuristic |
| Saidi-Mehrabad et al. (2015) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | | ✓ | MILP-TS, Ant Colony |
| Ahmadi-Javid and Hooshangi-Tabrizi (2015) | | ✓ | | | | ✓ | | | | ✓ | ✓ | | ✓ | | ✓ | MILP, ASO |
| Umar et al. (2015) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | | ✓ | Multi-objective GA |
| Bürgy and Gröflin (2016) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | ✓ | | rail-bound transportation, TS |
| Baruwa and Piera (2016) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | | ✓ | | Petri net-based heuristic |
| Nouri, Driss, and Ghédira (2016b) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | | ✓ | Holonic multiagent |
| Liu and Kozan (2017) | | | | ✓ | | | ✓ | ✓ | | | | | | | ✓ | Hybrid TS |
| Gultekin, Coban, and Akhlaghi (2018) | | | ✓ | | | | ✓ | ✓ | | | | | ✓ | | | MILP |
| Petrović, Miljković, and Jokić (2019) | | | | ✓ | | | ✓ | ✓ | | | | | | | | WO |
| Fontes and Homayouni (2019) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | | | MILP |
| Samarghandi (2019) | | | ✓ | | | | ✓ | | ✓ | | ✓ | | ✓ | | ✓ | MILP, CP, TS |
| Ham (2021) | | | | ✓ | | | ✓ | | | ✓ | ✓ | | ✓ | | | CP |
| Harrabi, Driss, and Ghedira (2021) | | | | ✓ | | | ✓ | | ✓ | | ✓ | | | | ✓ | Hybrid BBO |
| Boyer et al. (2021) | | | ✓ | | | | ✓ | | ✓ | | ✓ | | ✓ | | ✓ | MILP, CP, GRASP |
| Current study | | | | ✓ | | | ✓ | | | ✓ | | ✓ | ✓ | | | MILP, CP, SB-B&B |

Note: **IP**: Integer programming; **MINLP:** Mixed-integer non-linear programming; **MILP:** Mixed-integer linear programming; **STW**: Slide time window; **HGA**: Hybrid genetic algorithm; **FBS:** Filtered beam search; **TS:** Tabu search; **HBB:** Heuristic branch-and-bound; **DES:** Discrete event simulation; **MP**: Mathematical programming; **CP**: Constraint programming; **SA**: Simulated annealing; **LS:** Local search; **AI:** Artificial immune; **ASO:** Anarchic society optimisation; **WO**: Whale optimisation; **BBO**: Biogeography-based optimisation; **GRASP:** Greedy randomised adaptive search procedure**; SB-B&B:** Structure-based branch-and-bound.

- The problem is modelled using MILP and CP.
- The adjustment of the MILP model for the homogenous case performs better than the existing MILP model for this case.
- For the first time, SB-B&B is applied to solve such an integrated problem with a limited number of transporters.
- Interesting practical insights on using a heterogeneous transportation fleet are presented.

The rest of the paper is organised as follows. Section 2 formally states the problem and presents the MILP and CP models for the problem. Section 3 applies SB-B&B to solve the problem. Section 4 evaluates the solution methods. Section 5 provides managerial implications derived from the study. Finally, Section 6 concludes the paper.

## 2. The proposed scheduling problem

The first part of this section formally states our problem $J|T(free, lim, hetero)|C_{max}$ that integrates the scheduling of machines and transporters in a job-shop environment with a limited number of free and heterogeneous transporters (Section 2.1). Then, the proposed problem is modelled using MILP and CP (Sections 2.2 and 2.3).

### 2.1. Our problem

Our problem is defined in a job-shop environment. Job-shop scheduling is still an important area of research because of its various applications in different fields (Márquez and Ribeiro 2022; Xiong et al. 2022). Recent publications continue to demonstrate a keen interest in solving the classic job-shop scheduling problem using traditional methods (Gao et al. 2019; Liang et al. 2021; Xie et al. 2022). Moreover, complex job-shop scheduling problems are now studied and tackled by data-driven and machine learning methods (Park, Jeon, and Noh 2022; and Shady et al. 2022).

To define our integrated job-shop scheduling problem, consider a set of $n_{Jo}$ jobs $J = \{J_1, J_2, \ldots, J_{n_{Jo}}\}$ on a set of $n_{Ma}$ machines $M = \{M_1, M_2, \ldots, M_{n_{Ma}}\}$, transported by a set of $n_{Tr}$ heterogeneous transporters $T = \{T_1, T_2, \ldots, T_{n_{Tr}}\}$ with different transportation speeds, $sp_t, t \in T$. Each job $J_j$ consists of a sequence of $n_j$ operations $O_{j,1}, O_{j,2}, \ldots, O_{j,n_j}$, indicating the processing route of the job $J_j$. Each operation $O_{j,i}$ $(i = 1, \ldots, n_j)$ is processed on a specific machine. The aim is to determine the start and finish time spots of both processing operations and transportation tasks, together with the trip-assignments to the transporters such that the maximum completion time (makespan) is minimised. The other main assumptions of our problem are listed below:

1. At time zero there is a set of jobs with known processing routes.
2. A job-shop layout is given for the machines for which input/output buffers are sufficiently available.
3. Only one machine can process each job at a time instant, and only one job can be processed on each machine.
4. Jobs cannot be preempted, and their processing times are assumed to be deterministic.
5. Unprocessed jobs are distributed from and gathered in a single station, called load/unload (L/U) station.
6. Each transporter carries only one job at any time instant and starts its first trip from the L/U station.
7. Each operation of a job requires a *loaded* trip, which brings the job from the machine that processes the operation's predecessor. It also needs an *unloaded* trip (deadhead or empty moving), in which the transporter (allocated to carry the job) moves from its previous stop point to the place of the machine used for the operation's predecessor to pick up the job.
8. The moving speeds of the transporters can vary, resulting in transporter-dependent durations for transportation trips. The triangle inequality is hold for the distances among any three stop points. Jobs and (loaded or unloaded) trips do not impact on the moving speeds of the transporters.

Figure 1 presents an example of a job shop that includes four machines and three jobs with specific processing routes. There are five transporters where the speeds of the transporters in the two groups of $T_1, T_2, T_3$

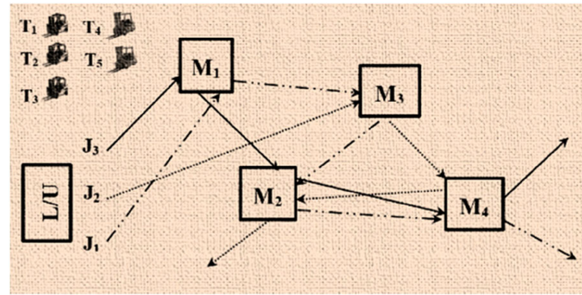



**Figure 1.** A job-shop with two types of transporters.

and $T_4, T_5$ are not the same due to their technological differences. All the transporters are located in the L/U station at time zero.

### 2.2. MILP formulation

MILP is a powerful method for modelling and solving various decision problems. This subsection formulates our problem as an MILP model, which can be solved by existing powerful commercial solvers.

In the case of heterogeneous transporters, transportation times between machines are not predetermined and their values directly depend on the speeds of the assigned transporters, which complicates the formulation. Table 2 presents the notation of the formulation. The MILP model for the $J|T(free, lim, hetero)|C_{max}$ problem can be written as follows:

$$\min CT^{Jo}_{max} \tag{1}$$

subject to:

$$CT^{Jo}_{max} \geq ST^{Op}_i + p_i \quad j \in J, i \in I : i = N_j + n_j \tag{2}$$

$$ST^{Op}_i - ST^{Op}_{i-1} \geq p_{i-1} + \sum_{t \in T} (d_{m(i-1),m(i)} \times z_{i,t})/sp_t \quad j \in J, i \in I_j : i \geq N_j + 2 \tag{3}$$

$$ST^{Op}_i \geq \sum_{t \in T} (d_{L/U,m(i)} \times z_{i,t})/sp_t \quad j \in J, i \in I : i = N_j + 1 \tag{4}$$

$$ST^{Op}_i \geq ST^{Op}_h + p_h - H \times q_{i,h} \quad i, h \in I : m(i) = m(h), i \neq h \tag{5}$$

$$q_{i,h} + q_{h,i} = 1 \quad i, h \in I : m(i) = m(h), i \neq h \tag{6}$$

$$\sum_{t \in T} z_{i,t} = 1 \quad i \in I \tag{7}$$

$$x_{0,i,t} + \sum_{h \in I^{es}_i} x_{h,i,t} = z_{i,l} \quad i \in I, t \in T \tag{8}$$

$$x_{h,0,t} + \sum_{i \in I^{ep}_h} x_{h,i,t} = z_{h,t} \quad i \in I, t \in T \tag{9}$$

**Table 2.** Sets, parameters, and variables used in the MILP model for $J|T(free, lim, hetero)|C_{max}$.

| Notation | Description |
|---|---|
| **Sets and parameters** | |
| $J$ | Set of available jobs, $\lvert J\rvert = n_{Jo}$ |
| $M$ | Set of processing machines, $\lvert M\rvert = n_{Ma}$ |
| $T$ | Set of transporters, $\lvert T\rvert = n_{Tr}$ |
| $n_j$ | Number of operations of job $j, j \in J$ |
| $N_j$ | Total number of operations of any job $k$ with $k < j$, i.e. $N_j = \sum_{k<j} n_k,\ N_1 = 0,\ j \in J$ |
| $I_j$ | Set of indices of operations of job $j$, i.e. $I_j = \{N_j + 1, \ldots, N_j + n_j\},\ j \in J$ |
| $I$ | Set of indices of all operations, i.e. $I = \bigcup_{j\in J} I_j = \{1, \ldots, N\}$ |
| $N$ | Total number of all operations, i.e. $N = \sum_{j\in J} n_j = \lvert I\rvert$ |
| $m(i)$ | Machine processing operation $i, i \in I$ |
| $j(i)$ | Job associated with operation $i, i \in I$ |
| $pre_i$ | Preceding operation of operation $i, i \in I$ |
| $I_i^{es}$ | Set of indices of operations of job $j(i)$ excluding operations with $k \geq i$, i.e. $I\backslash\{k \in I_{j(i)} \mid k \geq i\}, i \in I$ |
| $I_i^{ep}$ | Set of indices of operations of job $j(i)$ excluding operations with $k \leq i$, i.e. $I\backslash\{k \in I_{j(i)} \mid k \leq i\}, i \in I$ |
| $sp_t$ | Moving speed of transporter $t, t \in T$ |
| $d_{mm'}$ | Distance between two machines $m$ and $m'$, one of which can be L/U station |
| $H$ | A positive constant that is sufficiently large, i.e. $H \geq \sum_{i\in I} p_i + \sum_{j\in J}\left(n_j \times \min_{m,m'\in M}(d_{mm'})\right)$ |
| **Non-negative variables** | |
| $CT_i^{Tr}$ | Completion time of loaded transportation trip of operation $i, i \in I$ |
| $ST_i^{Op}$ | Start time of operation $i, i \in I$ |
| $CT_{max}^{Jo}$ | Makespan of all jobs |
| **Binary variables** | |
| $q_{h,i}$ | 1 if operation $h$ precedes operation $i$; $h, i \in I$ |
| $z_{i,t}$ | 1 if transporter $t$ delivers job to machine $m(i)$ to start operation $i$; $i \in I, t \in T$ |
| $x_{h,i,t}$ | 1 if transporter $t$ delivers job from $m(pre_i)$ to $m(i)$ after delivering job to machine $m(h)$; $i, h \in I, t \in T$ |
| $x_{0,i,t}$ | 1 if transporter $t$ delivers job from $m(pre_i)$ to $m(i)$ after being initialised from $\mathrm{L\backslash U}$; $i \in I, t \in T$ |
| $x_{h,0,t}$ | 1 if transporter $t$ performs last unloaded trip to $\mathrm{L\backslash U}$ after delivering job to machine $m(h)$; $h \in I, t \in T$ |

$$\sum_{i\in I} x_{0,i,t} \leq 1 \quad t \in T \tag{10}$$

$$\sum_{i\in I} x_{0,i,t} - \sum_{h\in I} x_{h,0,t} = 0 \quad t \in T \tag{11}$$

$$CT_i^{Tr} \leq ST_i^{Op} \quad i \in I \tag{12}$$

$$CT_i^{Tr} - \sum_{t\in T}(d_{m(i-1),m(i)} \times z_{i,t})/sp_t \geq ST_{i-1}^{Op} + p_{i-1} \qquad j \in J, i \in I_j : i \geq N_j + 2 \tag{13}$$

$$CT_h^{Tr} \geq (d_{L\backslash U,m(h)}/sp_t) + H \times (x_{0,h,t} - 1) \qquad t \in T, j \in J, h = N_j + 1 \tag{14}$$

$$CT_h^{Tr} \geq (d_{L/U,m(h-1)}/sp_t) + (d_{m(h-1),m(h)}/sp_t) + H \times (x_{0,h,t} - 1) \qquad t \in T, j \in J, h \in I_j, h \neq N_j + 1 \tag{15}$$

$$CT_h^{Tr} \geq CT_i^{Tr} + (d_{m(i),L/U}/sp_t) + (d_{L/U,m(h)}/sp_t) + H \times (x_{i,h,t} - 1) \qquad t \in T, i \in I, j \in J, h = N_j + 1, h \neq i \tag{16}$$

$$CT_h^{Tr} \geq CT_i^{Tr} + (d_{m(i),m(h-1)}/sp_t) + (d_{m(h-1),m(h)}/sp_t) + H \times (x_{i,h,t} - 1) \qquad t \in T, i \in I, j \in J, h \in I_j h \neq N_j + 1, h \neq i \tag{17}$$

$$ST_i^{Op} \geq 0,\ CT_i^{Tr} \geq 0,\ CT_{max}^{Jo} \geq 0 \quad i \in I \tag{18}$$

$$q_{i,h} \in \{0, 1\} \quad i, h \in I : m(i) = m(h) \tag{19}$$

$$z_{i,t} \in \{0, 1\} \quad i \in I, l \in T \tag{20}$$

$$x_{h,i,l} \in \{0, 1\} \quad i, h \in I \cup \{0\}, t \in T \tag{21}$$

Objective (1) is the minimisation of the makespan. Constraint set (2) guarantees that the end times of the last operations of all the jobs are less than or equal to the makespan. Constraint sets (3) and (4) set up the precedence relations among processing and transportation operations. Constraint sets (5) and (6) are machine-disjunction constraints, which specify the sequences of operations performing on each machine. Constraint set (7) ensures that each operation is allocated to only one transporter. Constraint sets (8) and (9) guarantee that for each transporter assigned to any operation only one loaded trip and only one unloaded trip can be considered.

Constraint set (10) controls the number of transporters being initialised from the L/U station, that is, each transporter must enter the system at most once. Constraint set (11) guarantees that the transporters distributed in the job-shop environment must be gathered in the L/U station after completing their last tasks. In other words, a transporter starts from the L/U station by performing its first unloaded trip, and after finishing all the assigned trips, it makes a final unloaded trip to return to the L/U station.

Constraint set (12) forces that any operation of a job can be processed only if the job is delivered to the corresponding machine. Similarly, Constraint set (13) enforces that only after the completion of the preceding operation, a loaded trip can be done. Note that considering (12) and (13) one can drop constraint set (3), which is kept here for better understanding the model. Constraint sets (14)–(17) are transporter-disjunction constraints, which specify the sequences of intermediate transportation trips allocated to each transporter. Constraint sets (18)–(21) are domain constraints.

In the above formulation, the number of variables and constraints is a polynomial function of the number of jobs and the number of machines; thus, the model can be solved using existing powerful MILP solvers.

### 2.3. CP modelling

This subsection presents a CP model for our problem. CP has been recently used for solving difficult combinatorial optimisation problems. There are some specialised variables and constraints defined in CP that facilitate modelling of scheduling problems. *Interval* and *sequence* variables are two useful types of decision variables in CP that are used to formulate scheduling problems. An interval variable $x$ represents a time interval during which a job is being processed. Interval variables are characterised by the start time ($Start(x)$), end time ($End(x)$), processing time, and presence status ($Presence(x)$), where $Start(x)$, $End(x)$, and $Presence(x)$ are endogenous decisions. Interval variables may be defined as optional or present. If $x$ is defined as present, all feasible solutions must include the interval variable and satisfy $Presence(x) = 1$. If it is defined as optional, model decides if the variable is present in the solution or not. Optional interval variables that are used for the assignment of jobs to the machines are endogenous decisions. Specialised constraint *Alternative* is used to assign a job to exactly one machine by choosing only one of the associated optional interval variables that are defined for the job. Sequence variables provide a permutation for a set of interval variables that are present in the solution and are used to define the processing sequence of a set of jobs on the same machine. Sequence variables define a sequence of interval variables without enforcing any constraint on the relative position of the intervals in the temporal dimension. To enforce that consecutive interval variables do not overlap in time, one can use specialised constraints *NoOverlap*.

Using the variables defined in Table 3, our problem can be represented by the following CP model:

$$\min \quad CT^{Jo}_{max} \tag{22}$$

subject to:

$$CT^{Jo}_{max} \geq End(JobOnMch_i) \quad j \in J, i \in I : i = N_j + n_j \tag{23}$$

$$EndBeforeStart(JobOnMch_{i-1}, JobOnMch_i) \quad j \in J, i \in I_j : i \geq N_j + 2 \tag{24}$$

$$EndBeforeStart(JobOnMch_{i-1}, Drop_i) \quad j \in J, i \in I_j : i \geq N_j + 2 \tag{25}$$

$$EndBeforeStart(Drop_i, JobOnMch_i) \quad i \in I \tag{26}$$

$$Alternative(Drop_i, \{Move4Drop_{i,t}\}_{t \in T}) \quad i \in I \tag{27}$$

$$NoOverlap(SeqMch_m) \quad m \in M \tag{28}$$

$$NoOverlap(SeqTr_t, trans_t) \quad t \in T \tag{29}$$

Objective (22) is the minimisation of the makespan. Constraint set (23) guarantees that the end times of the last operations of all the jobs are less than or equal to the makespan. Constraint set (24) enforces the precedence relations among the processing operations associated with each job. Constraint set (25) ensures that only after the completion of each preceding operation, the loaded trip can be done. Constraint set (26) forces that any operation of a job can be processed only if the job is delivered to the corresponding machine. Constraint set (27) allocates each operation to exactly one transporter. Constraint set (28) guarantees that on each machine, the processing of associated operations does not overlap in time. Constraint set (29) ensures that not only the delivery tasks assigned to transporter $t$ does not overlap in

**Table 3.** Variables used in the CP model for $J|T(free, lim, hetero)|C_{max}$.

| Notation | Description |
|---|---|
| $Drop_i$ | Interval variable denoting the actual time interval for delivering operation $i$ from $m(pre_i)$ to $m(i)$, $i \in I$ |
| $Move4Drop_{i,t}$ | Optional interval variable denoting the time interval for delivering operation $i$ from $m(pre_i)$ to $m(i)$ if it is assigned to transporter $t$; $i \in I, t \in T$ |
| $JobOnMch_i$ | Interval variable representing the actual time interval for processing operation $i$ on $m(i)$, $i \in I$ |
| $SeqMch_m$ $\{JobOnMch_i, i \in I^M_m\}$ | Sequence variable keeping the sequence of all operations that must be processed on machine $m$; it is defined on interval variables $JobOnMch_i$, $i \in I^M_m$ ($I^M_m$ denotes the set of operations with $m(i) = m$); $m \in M$ |
| $SeqTr_t\{Move4Drop_{i,t}, i \in I, t \in T\}$ | Sequence variable keeping the sequence of all operations that must be delivered by transporter $t$; it is defined on interval variables $Move4Drop_{i,t}$, $i \in I$, that are present in the solution, i.e. $SeqTr_t$ is the sequence of $Move4Drop_{i,t}$ with $Presence(Move4Drop_{i,t}) = 1, i \in I; t \in T$ |

time, but also there is a predefined time between the end time of a trip and the start time of the following trip, specified by the transition matrix $trans_t$, which depends on the speed of the transporter $t$. For each transporter $t \in T$, $trans_t$ is an $N \times N$ input matrix whose elements are calculated as follows:

$$trans_t(h,i) = \frac{d_{m(h)} + d_{m(i-1)}}{sp_t}$$

for each pair of operations $h, i \in I$.

## 3. Branch-and-bound

Branch-and-bound (B&B) is a key method to solve many challenging combinatorial optimisation problems. Since most of the integrated scheduling problems are often highly constrained and complicated, B&B algorithms often fail to solve them unless there is a special attention to exploit their solution structures. In the literature, there are some studies that have applied B&B instead of using the mathematical models to solve flow-shop and job-shop scheduling problems (Brucker, Jurisch, and Sievers 1994; Chen 2006; Chung, Flynn, and Kirca 2006). This section shows how our problem (and the traditional simpler problems) can be solved by a B&B algorithm.

Our B&B algorithm is directly developed based on the sequence of operations (it is not LP-based and is independent of any problem's formulation). In the following, this algorithm is called SB-B&B (Structure-Based B&B) as it uses the structure of the solutions of the problem, specifically the job sequence here.

### 3.1. Algorithm description

The proposed SB-B&B algorithm is based on the sequence of operations ($O_{j,i}, i = 1, 2, \ldots, n_j, j = 1, 2, \ldots, n_{Jo}$). In this algorithm, a search tree with $N$ levels ($N = |I|$) is created. At level $l$ ($l = 0, 1, \ldots, N$), each node of the search tree is associated to a partial sequence of operations $\sigma_l$ in which $l$ operations of jobs are sequenced and $N - l$ operations remain unscheduled. The algorithm starts with a root node ($l = 0$) having no operation sequenced. The first branching assigns an operation of a job to the first available position and the second branching can sequence either the next operation of the same job or the first operation of a different job in the second available position. This procedure continues until all operations are sequenced. In other words, at each level, an operation of a job is scheduled considering the technological route of the operations belonging to each job.

The suggested algorithm uses the Depth-First Search (DFS) and backtracking strategies. According to the DFS strategy, the algorithm starts from the root (partial sequence of zero operations) and explores as far as possible along each branch before backtracking. At each level, a node with the smallest lower bound is chosen to keep the searching procedure. After reaching to the deepest level of the search tree, the search backtracks and returns to the most recent node and continues to explore other sub-nods. The DFS strategy helps us to find a very quick feasible solution. Besides, it needs less memory to explore the tree. At each node of the tree, a lower bound for the associated partial sequence is calculated and a dominance rule can be also applied to fathom the nodes and reduce the dimension of the search space.

### 3.2. Lower-bounding strategies

In this subsection, two lower-bounding strategies are proposed and explained in detail. They can be used to find lower bounds for the partial schedules created by the search tree.

#### 3.2.1. Lower-bounding strategy $LB_1$

Our first lower-bounding strategy considers an instance of $J/T/C_{max}$ and relaxes the restriction imposing that each machine can process at most one operation at any time instant. Furthermore, the minimum moving time (maximum moving speed) is assumed for each transporter. Therefore, a release date for each operation $i$ can be calculated as follows:

$$r_i = \sum_{k \in P(i)} \left( \frac{1}{\max_{t \in T}\{sp_t\}} d_{m(pre_k),m(k)} + p_k \right) + \frac{1}{\max_{t \in T}\{sp_t\}} d_{m(pre_i),m(i)} \quad i \in I,$$

where $P(i) = \{N_j + 1, \ldots, pre_i\} \subseteq I_{j(i)}$ is the set of all preceding operations of operation $i$ belonging to the same job. This relaxation of the problem yields a one-machine scheduling problem with release dates, denoted by $1/r_i, T/C_{max}$. This provides a lower bound for each machine as follows:

$$lb_{1,m} = \min_{i \in O_m}(r_i) + \sum_{i \in O_m} p_i \quad m \in M$$

where $O_m$ is the set of operations associated with machine $m$. Therefore, a valid lower bound for our problem is given by

$$\mathrm{LB}_1 = \max_{m \in M}(lb_{1,m})$$

#### 3.2.2. Lower-bounding strategy $LB_2$

The second relaxation of the problem, denoted by $1/r_i, q_i, T/C_{max}$, can be obtained by setting the release dates used in $\mathrm{LB}_1$ and defining delivery times as follows:

$$q_i = \frac{1}{\max\limits_{t\in T}\{sp_t\}} d_{m(i),m(fol_i)} + \sum_{k\in F(i)} \left( p_k + \frac{1}{\max\limits_{t\in T}\{sp_t\}} d_{m(k),m(fol_k)} \right) \quad i \in I$$

where $F(i) = \{fol_i, \ldots, N_j + n_j\} \subseteq I_{j(i)}$ is the set of all succeeding operations of operation $i$, belonging to the same job, and where $fol_i$ denotes the operation that immediately follows operation $i$. The resulting relaxation is also a one-machine problem, but with both release dates and delivery times. Hence, for each machine the following lower bound:

$$lb_{2,m} = \min_{j\in O_m}\{r_i\} + \sum_{i\in O_m} p_i + \min_{i\in O_m}\{q_j\} \quad m \in M$$

can be obtained, in which the release dates and delivery times of all the operations are set to $\min\limits_{i\in O_m}(r_i)$ and $\min\limits_{i\in O_m}(q_i)$, respectively. This yields another lower bound for our problem as

$$\mathrm{LB}_2 = \max_{m\in M}(lb_{2,m}).$$

It is clear that the second lower bound is always better than the first, $\mathrm{LB}_2 \geq \mathrm{LB}_1$, but its computing requires slightly more efforts. Developing other possible lower bounds remains as a future study.

Now let us show how each lower-bounding strategy can be adjusted in our SB-B&B algorithm. The procedure is explained for the second one, which can be adjusted for the first with small changes. For a given partial sequence of operations (obtained at each node), to find a lower bound based on the second lower-bounding strategy, one can use the following procedure:

**Initialise** $ST_i^{Op} = 0;\ CT_i^{Op} = 0,\ i \in I;\ C_m^{Ma} = 0,\ m \in M;\ C_t^{Tr} = 0,\ L_t = \mathrm{L/U},\ t \in T$
**For** each scheduled operation $i$
  Compute $CT_i^{Op}$
**endfor**
**Compute** $C_m^{Ma} = \max\limits_{i\in O_m}(CT_i^{Op})$ for each machine $m \in M$
**For** each unscheduled operation $i$ do
  **If** $pre_i = 0$
   $r_i = \frac{1}{\max\limits_{t\in T}\{sp_t\}} d_{L/U,m(i)}$
  **else**
   $r_i = \sum\limits_{k\in P(i)} \left( \frac{1}{\max\limits_{t\in T}\{sp_t\}} d_{m(pre_k),m(k)} + p_k \right) + \frac{1}{\max\limits_{t\in T}\{sp_t\}} d_{m(pre_i),m(i)}$
  **endif**
  $q_i = \frac{1}{\max\limits_{t\in T}\{sp_t\}} d_{m(i),m(fol_i)} + \sum\limits_{k\in F(i)} \left( p_k + \frac{1}{\max\limits_{t\in T}\{sp_t\}} d_{m(k),m(fol_k)} \right)$
**endfor**
**For** each $m \in M$ do
  $MinR_m = \max\left\{ C_m^{Ma}, \min\limits_{i\in O_m} r_i \right\},\ MinQ_m = \min\limits_{i\in O_m} q_i$
  $lb_{2m} = MinR_m + \sum\limits_{i\in O_m} p_i + MinQ_m$
**endfor**
**Return** $\mathrm{LB}_2 = \max\limits_{m\in M}(lb_{2m})$

where $CT_i^{Op}$, $C_m^{Ma}$, and $C_t^{Tr}$ denote the completion times of operation $i \in I$, machine $m \in M$, and transporter $t \in T$, respectively, and where $L_t$ denotes the last visited position for transporter $t$ (the L/U station or the place of a machine can represent a position here). Note that if $C_m^{Ma} = 0$ for all machines, the resulting lower bound becomes identical to the first lower bound. In fact, at each node one should update only $MinR_m$ for all machines.

In the fourth line of the above pseudo code, to compute the makespan $CT_i^{Op}$ for each scheduled operation $i$, one can use the adjusted MILP model (Section 2.2) or CP model (Section 2.3), respectively, after an appropriate resetting (fixing the values of the operation-assignment variables in the MILP model and adapting constraints (24) in the CP model) such that the resulting operation sequence becomes identical to the partial sequence (these adjusted models only schedule the transports while the sequence of the jobs is given). This method is the most effective one, but can be very time consuming.

Another method is to use a lower bound for $CT_i^{Op}$ that can be computed efficiently; however, finding such a lower bound that effectively uses the given partial job sequence and some characteristics of the transportation fleet at each node seems difficult (and remains as an open problem). In fact, there are problem instances for which the makespan of the partial sequence with consideration of the transportation times equals to the makespan without considering the transportation times plus the minimum of the transportation times of the first jobs on all the machines using the fastest transporter (which means that the latter is a tight lower bound for the makespan with consideration of the transportation times). If the triangular inequality holds for the distances, another lower bound is the maximum of all the lower bounds that can be computed for each machine, i.e. the makespan of the machine (without considering transportation times) plus the transportation time of its first job using the fastest transporter. This lower bound is also tight. One may extend the existing lower bounds for permutation flowshops with minimal time lags (Fondrevelle, Oulamara, and Portmann 2006; Hamdi and Loukil 2015) for job shops with minimal time lags (that are set to the transportation times required by the fastest transporter).

Another method is to approximate $CT_i^{Op}$ using the following greedy algorithm in which the best transport is always selected for each transportation task:

**Set** $ST_i^{Op} = CT_{pre_i}^{Op}$
**Find** the nearest transporter $BestTr$ to $m(pre_i)$, i.e.
$BestTr \in \underset{t \in T}{argmin}\{d_{L_t,m(pre_i)}\}$
**If** $CT_{pre_i}^{Op} > C_{BestTr}^{Tr}$
$C_{BestTr}^{Tr} = CT_{pre_i}^{Op}$
**endif**
$C_{BestTr}^{Tr} = C_{BestTr}^{Tr} + (d_{L_t,m(pre_i)} + d_{m(pre_i),m(i)})/sp_t$
**Update** the last visited position of $BestTr$, i.e.
$L_{BestTr} = m(i)$
**If** $ST_i^{Op} < C_{BestTr}^{Tr}$
$ST_i^{Op} = C_{BestTr}^{Tr}$
**endif**
**If** $ST_i^{Op} < C_{m(i)}^{Ma}$
$ST_i^{Op} = C_{m(i)}^{Ma}$
**endif**
**Return** $C_{m(i)}^{Ma} = ST_i^{Op} + p_i$, $CT_i^{Op} = C_{m(i)}^{Ma}$

for each $i \in I$ where $ST_i^{Op} = 0$, $CT_i^{Op} = 0$, $C_m^{Ma} = 0$, $C_t^{Tr} = 0$, $L_t =$ L/U for all $i \in I, m \in M, t \in T$.

In the sequel, the SB-B&B algorithm is called M-SB-B&B or C-SB-B&B if the adjusted MILP or CP model is used to compute the makespan $CT_i^{Op}$, and A-SB-B&B if the above procedure is used.

For a given complete scheduling at a final B&B node, one should compute the makespan with consideration of transportation times (the upper bound). To compute the upper bound, one can use one of the MILP and CP models after fixing the sequence of jobs on each machine. Instead, one can use the greedy algorithm described above for more efficiency (as it is done here) though the optimality of some solutions cannot be proven.

### 3.3. Dominance rule

A dominance rule can be used for comparing partial sequences of operations in the B&B algorithm, which helps the algorithm prune the nodes faster during the traversing process. Let $\sigma_l^1$ and $\sigma_l^2$ be two partial sequences of size $l$ whose sets of operations are the same $S$ ($S \subset I, |S| = l$) and let $\sigma$ be any permutation of the operations in $I \backslash S$ (unscheduled operations). If for any $\sigma$, $C_{max}(\sigma_l^1, \sigma) \geq C_{max}(\sigma_l^2, \sigma)$, then $\sigma_l^2$ *dominates* $\sigma_l^1$, or $\sigma_l^1$ is *dominated* by $\sigma_l^2$. It should be noted that $C_{max}(\sigma_l^i, \sigma), i = 1, 2,$ is the makespan obtained by the complete solution $\sigma_l^i, \sigma$ with $N$ scheduled operations. A dominance rule for our problem is given below.

**Theorem 3.1:** *Let $\sigma_l^1$ and $\sigma_l^2$ be two partial sequences with the same set of operations S. $\sigma_l^1$ is dominated by $\sigma_l^2$ if*

$$C_{max}(\sigma_l^1) \geq C_{max}(\sigma_l^2) + \max_{t \in T}\left\{\frac{1}{sp_t} d_{L_t,L_t'}\right\},$$

*where $L_t$ and $L_t'$ are the stop points of transporter $t \in T$ when the maximum completion times of the partial sequences $\sigma_l^1$ and $\sigma_l^2$ are reached, respectively.*

***Proof:*** Let $t_i^\sigma$ denote the completion time of the partial sequence $\sigma_l^i$, $i = 1, 2$. If at the time instant $t_1^\sigma$, all the transporters move from $L_t$ to $L_t'$, then the positional states become identical for both cases (each transporter $t$ is at the same stop point $L_t'$ after movement). The maximum time required for moving all the fleet is $\max_{t \in T}\{(1/sp_t)d_{L_t,L_t'}\}$. Hence, if $C_{max}(\sigma_l^2)$ plus this maximum moving time does not exceed $C_{max}(\sigma_l^1)$, then $\sigma_l^2$ obviously dominates $\sigma_l^1$. This completes the proof. ■

The use of Theorem 3.1 for arbitrary partial sequences seems impractical because of its excessive memory requirement. Instead, the following corollary can be used, which focuses on the special partial sequences differing on their last two operations.

**Corollary 3.1:** *Assume that $\sigma_l$ is an arbitrary partial sequence with operation set S ($|S| = l$), and let i and j be distinct operations in $I \backslash S$ that are supposed to be sequenced at the levels $l$ and $l + 1$, respectively. If*

$$C_{max}(\sigma_l, j, i) \geq C_{max}(\sigma_l, i, j) + \max_{t \in T}\left\{\frac{1}{sp_t} d_{L_t,L_t'}\right\},$$

*then the partial sequence $\sigma_l, i, j$ dominates the partial sequence $\sigma_l, j, i$, and the node associated with operation j can be fathomed ($L_t$ and $L_t'$ are the stop points of transporter $t \in T$ when the maximum completion times of the partial sequences $\sigma_l, j, i$ and $\sigma_l, i, j$ are reached, respectively).*

To compute the completion times $C_{max}(\sigma_l, j, i)$ and $C_{max}(\sigma_l, i, j)$, appearing in the above corollary, one can use one of the adjusted MILP and CP models for assign-

**Table 4.** Computational results for assessing the proposed solution methods.

| Instance ↓, Solution method → | | | | | MILP model | | CP Model | | A-SB-B&B ($LB_1$) | | A-SB-B&B ($LB_2$) | | M-SB-B&B ($LB_2$) | | C-SB-B&B ($LB_2$) | | ASO | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | #Operations | #Machines | #Jobs | #Transporters | Obj | time (s) | Obj | time (s) | Obj | time (s) | Obj | time (s) | Obj | time (s) | Obj | time (s) | Min | time (s) |
| Exp1-1 | 21 | 4 | 5 | 2 | **94.7** | 12.4 | **94.7** | 1.8 | **94.7** | 0.3 | **94.7** | 0.2 | 102.7 | 900.0 | 107.7 | 900.0 | **94.7** | 1.0 |
| Exp1-2 | 21 | 4 | 5 | 2 | **78.8** | 2.4 | **78.8** | 0.2 | **78.8** | 0.2 | **78.8** | 0.1 | 78.8 | 900.0 | 80.5 | 900.0 | **78.8** | 1.0 |
| Exp1-3 | 21 | 4 | 5 | 2 | **82.0** | 1.5 | **82.0** | 0.3 | **82.0** | 0.3 | **82.0** | 0.2 | 82.7 | 900.0 | 82.7 | 900.0 | 82.3 | 1.0 |
| Exp1-4 | 21 | 4 | 5 | 2 | **101.0** | 6.4 | **101.0** | 1.1 | **101.0** | 0.3 | **101.0** | 0.2 | 105.2 | 900.0 | 105.2 | 900.0 | 101.7 | 1.0 |
| Exp2-1 | 24 | 4 | 6 | 2 | **97.5** | 395.8 | **97.5** | 2.7 | **97.5** | 4.1 | **97.5** | 1.0 | 104.5 | 900.0 | 102.5 | 900.0 | **97.5** | 1.2 |
| Exp2-2 | 24 | 4 | 6 | 2 | **76.7** | 5.5 | **76.7** | 0.2 | **76.7** | 2.4 | **76.7** | 0.4 | 80.3 | 900.0 | 80.0 | 900.0 | 78.7 | 1.2 |
| Exp2-3 | 24 | 4 | 6 | 2 | **81.7** | 31.3 | **81.7** | 0.1 | **81.7** | 2.2 | **81.7** | 0.5 | 85.3 | 900.0 | ***81.7*** | 900.0 | **81.7** | 1.2 |
| Exp2-4 | 24 | 4 | 6 | 2 | **108.7** | 454.7 | **108.7** | 5.4 | **108.7** | 8.0 | **108.7** | 2.3 | 114.5 | 900.0 | ***108.7*** | 900.0 | 112.7 | 1.2 |
| Exp3-1 | 27 | 4 | 6 | 2 | **99.5** | 66.2 | **99.5** | 1.3 | 99.7 | 5.0 | 99.7 | 1.7 | 106.2 | 900.0 | 118.3 | 900.0 | 103.7 | 1.2 |
| Exp3-2 | 27 | 4 | 6 | 2 | **82.0** | 7.2 | **82.0** | 0.1 | **82.0** | 4.8 | **82.0** | 2.1 | 124.3 | 900.0 | 101.3 | 900.0 | 82.6 | 1.2 |
| Exp3-3 | 27 | 4 | 6 | 2 | **83.3** | 3.2 | **83.3** | 0.2 | **83.3** | 4.0 | **83.3** | 1.2 | 96.2 | 900.0 | 96.2 | 900.0 | **83.3** | 1.2 |
| Exp3-4 | 27 | 4 | 6 | 2 | ***110.0*** | 900.0 | **110.0** | 1.8 | **110.0** | 5.6 | **110.0** | 1.8 | 133.3 | 900.0 | 117.0 | 900.0 | 111.0 | 1.2 |
| Exp4-1 | 33 | 4 | 5 | 2 | 113.0 | 900.0 | **109.3** | 15.6 | **109.3** | 169.9 | **109.3** | 56.9 | 155.5 | 900.0 | 126.5 | 900.0 | 117.6 | 1.0 |
| Exp4-2 | 33 | 4 | 5 | 2 | 88.5 | 900.0 | **85.7** | 6.4 | **85.7** | 61.8 | **85.7** | 23.2 | 113.0 | 900.0 | 104.7 | 900.0 | 91.3 | 1.0 |
| Exp4-3 | 33 | 4 | 5 | 2 | 88.7 | 900.0 | **88.5** | 6.5 | **88.5** | 84.2 | **88.5** | 41.0 | 124.5 | 900.0 | 119.0 | 900.0 | 98.2 | 1.0 |
| Exp4-4 | 33 | 4 | 5 | 2 | 122.8 | 900.0 | **120.5** | 9.1 | **120.5** | 120.4 | **120.5** | 49.3 | – | 900.0 | 144.5 | 900.0 | 131.2 | 1.0 |
| Exp5-1 | 21 | 4 | 5 | 2 | **87.0** | 60.3 | **87.0** | 3.7 | **87.0** | 0.6 | **87.0** | 0.4 | 93.0 | 900.0 | 88.7 | 900.0 | **87.0** | 1.0 |
| Exp5-2 | 21 | 4 | 5 | 2 | **68.0** | 5.6 | **68.0** | 0.7 | **68.0** | 0.3 | **68.0** | 0.2 | 70.0 | 900.0 | 79.7 | 900.0 | 68.3 | 1.0 |
| Exp5-3 | 21 | 4 | 5 | 2 | **69.0** | 4.3 | **69.0** | 0.5 | **69.0** | 0.4 | **69.0** | 0.3 | 79.7 | 900.0 | 79.0 | 900.0 | 69.3 | 1.0 |
| Exp5-4 | 21 | 4 | 5 | 2 | **94.0** | 36.7 | **94.0** | 2.8 | **94.0** | 0.6 | **94.0** | 0.3 | 99.0 | 900.0 | ***94.0*** | 900.0 | 94.2 | 1.0 |
| Exp6-1 | 30 | 4 | 6 | 2 | **113.5** | 230.2 | **113.5** | 4.1 | **113.5** | 70.7 | **113.5** | 10.0 | 133.7 | 900.0 | 119.3 | 900.0 | 119.6 | 1.2 |
| Exp6-2 | 30 | 4 | 6 | 2 | **98.0** | 27.8 | **98.0** | 1.8 | **98.0** | 72.0 | **98.0** | 7.3 | 103.3 | 900.0 | 102.3 | 900.0 | 99.6 | 1.2 |
| Exp6-3 | 30 | 4 | 6 | 2 | **100.5** | 33.6 | **100.5** | 1.9 | **100.5** | 98.4 | **100.5** | 11.8 | 125.0 | 900.0 | 108.0 | 900.0 | 102.7 | 1.2 |
| Exp6-4 | 30 | 4 | 6 | 2 | 121.3 | 900.0 | **119.2** | 6.7 | 119.7 | 53.5 | 119.7 | 6.0 | 140.3 | 900.0 | 132.5 | 900.0 | 126.7 | 1.2 |
| Exp7-1 | 31 | 4 | 8 | 2 | 115.7 | 900.0 | **111.0** | 851.3 | 112.0 | 900.0 | ***111.0*** | 900.0 | – | 900.0 | 120.3 | 900.0 | 119.7 | 1.6 |
| Exp7-2 | 31 | 4 | 8 | 2 | 78.0 | 900.0 | **77.7** | 3.8 | 77.7 | 550.0 | **77.7** | 50.6 | 88.3 | 900.0 | 89.7 | 900.0 | 87.2 | 1.6 |
| Exp7-3 | 31 | 4 | 8 | 2 | 84.3 | 900.0 | **84.2** | 7.0 | 85.3 | 900.0 | **84.2** | 231.6 | 105.2 | 900.0 | 93.3 | 900.0 | 91.3 | 1.6 |
| Exp7-4 | 31 | 4 | 8 | 2 | 129.2 | 900.0 | **125.5** | 478.2 | ***125.5*** | 900.0 | ***125.5*** | 900.0 | – | 900.0 | 140.0 | 900.0 | 134.3 | 1.6 |
| Exp8-1 | 34 | 4 | 6 | 2 | **157.7** | 35.0 | **157.7** | 0.2 | ***157.7*** | 900.0 | **157.7** | 0.2 | **157.7** | 49.0 | **157.7** | 26.2 | 158.7 | 1.2 |
| Exp8-2 | 34 | 4 | 6 | 2 | **149.3** | 12.9 | **149.3** | 0.1 | 173.3 | 900.0 | **149.3** | 0.2 | **149.3** | 40.4 | ***149.3*** | 900.0 | **149.3** | 1.2 |
| Exp8-3 | 34 | 4 | 6 | 2 | **151.0** | 10.2 | **151.0** | 0.1 | ***151.0*** | 900.0 | **151.0** | 0.0 | **151.0** | 7.5 | ***151.0*** | 900.0 | 151.2 | 1.2 |
| Exp8-4 | 34 | 4 | 6 | 2 | ***161.0*** | 900.0 | **161.0** | 2.9 | ***161.0*** | 900.0 | **161.0** | 9.1 | ***161.0*** | 900.0 | ***161.0*** | 900.0 | 163.3 | 1.2 |
| Exp9-1 | 29 | 4 | 5 | 2 | **113.0** | 43.0 | **113.0** | 1.3 | 114.5 | 6.8 | 114.5 | 0.7 | 121.5 | 900.0 | 131.0 | 900.0 | 117.7 | 1.0 |
| Exp9-2 | 29 | 4 | 5 | 2 | **101.0** | 18.4 | **101.0** | 0.5 | 101.3 | 4.3 | 101.3 | 0.4 | 105.3 | 900.0 | 133.8 | 900.0 | 105.2 | 1.0 |
| Exp9-3 | 29 | 4 | 5 | 2 | **104.7** | 24.7 | **104.7** | 0.9 | **104.7** | 8.4 | **104.7** | 0.5 | 144.5 | 900.0 | 122.0 | 900.0 | 107.3 | 1.0 |
| Exp9-4 | 29 | 4 | 5 | 2 | **118.0** | 44.4 | **118.0** | 2.1 | 119.5 | 5.4 | 119.5 | 0.6 | 123.7 | 900.0 | 130.7 | 900.0 | 123.3 | 1.0 |
| Exp10-1 | 36 | 4 | 6 | 2 | 149.7 | 900.5 | **144.3** | 5.2 | 145.7 | 900.0 | **144.3** | 397.9 | 159.7 | 900.0 | 181.7 | 900.0 | 154.2 | 1.2 |
| Exp10-2 | 36 | 4 | 6 | 2 | **133.2** | 170.9 | **133.2** | 2.6 | 135.3 | 900.0 | 135.3 | 900.0 | 166.7 | 900.0 | 175.0 | 900.0 | 138.6 | 1.2 |
| Exp10-3 | 36 | 4 | 6 | 2 | 139.0 | 900.0 | **135.7** | 3.1 | ***135.7*** | 900.0 | ***135.7*** | 900.0 | 195.7 | 900.0 | 176.3 | 900.0 | 144.2 | 1.2 |
| Exp10-4 | 36 | 4 | 6 | 2 | 160.7 | 900.0 | **157.2** | 32.1 | 158.0 | 900.0 | 157.7 | 326.3 | 190.0 | 900.0 | 167.5 | 900.0 | 168.7 | 1.2 |

Note: The optimal objective values are indicated in bold font, and those that are optimal but not verified/finalized are shown in italic bold font

ing the transporters, but it is time consuming. An alternative is to approximate them by the greedy algorithm proposed earlier in Section 3.2.2. This idea is used in A-SB-B&B algorithm, but no dominance rule is used in M-SB-B&B and C-SB-B&B algorithms for the sake of efficiency.

## 4. Numerical study

In this section, the computational efficiency of the models and algorithms developed for $J|T(free, lim, hetero)|C_{max}$ is assessed. CPLEX 12.1 is used to solve the CP and MILP models, which are developed in Visual C++ using concert technology, on a personal computer with an Intel® Core™ i7-1065G7 CPU (1.5 GHz) and 16 GB RAM, operating Windows 11. Also, Visual C++ is used to implement all the proposed algorithms.

In our numerical study, the 40 slightly-adjusted test problems presented by Bilge and Ulusoy (1995) (created by 10 different job sets and 4 layouts of machines) are used where the speeds of the two transporters are set to 0.8 and 1.2. The configuration is reflected in the name of each instance. For example, Exp1-2 represents an experiment (Exp) considering the first job set (1) and the second machine layout (2).

Table 4 reports the results of the MILP model (Section 2.2), CP model (Section 2.3), and SB-B&B algorithms (Section 3) for the problem $J|T(free, lim, hetero)|C_{max}$. The upper bound is set to 500 in the B&B algorithms, and time limit is set to 900 s. All times are reported in seconds. The column 'Obj' repots the optimal or the best

**Table 5.** Computational results for the homogenous case.

| Solution method → | Old MILP model | | New MILP model | | Adjusted MILP model | | CP model | | A-SB-B&B ($LB_1$) | | A-SB-B&B ($LB_2$) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Instance ↓ | Obj | time (s) | Obj | time (s) | Obj | time (s) | Obj | time (s) | Obj | time (s) | Obj | time (s) |
| Exp1-1 | **96** | 2.4 | **96** | 9.6 | **96** | 1.3 | **96** | 1.3 | **96** | 0.5 | **96** | 0.3 |
| Exp1-2 | **82** | 0.7 | **82** | 3.2 | **82** | 0.4 | **82** | 0.1 | **82** | 0.5 | **82** | 0.1 |
| Exp1-3 | **84** | 2.5 | **84** | 4.0 | **84** | 0.5 | **84** | 0.1 | **84** | 0.6 | **84** | 0.2 |
| Exp1-4 | **103** | 4.4 | **103** | 18.3 | **103** | 1.1 | **103** | 1.8 | **103** | 0.7 | **103** | 0.2 |
| Exp2-1 | **100** | 39.2 | **100** | 22.2 | **100** | 5.9 | **100** | 1.9 | **100** | 7.5 | **100** | 1.2 |
| Exp2-2 | **76** | 6.4 | **76** | 3.0 | **76** | 1.0 | **76** | 0.1 | **76** | 0.9 | **76** | 0.1 |
| Exp2-3 | **86** | 20.9 | **86** | 18.6 | **86** | 2.2 | **86** | 0.9 | **86** | 6.1 | **86** | 0.8 |
| Exp2-4 | **108** | 106.3 | **108** | 441.6 | **108** | 15.7 | **108** | 2.3 | **108** | 11.3 | **108** | 1.7 |
| Exp3-1 | **99** | 20.8 | **99** | 87.8 | **99** | 3.2 | **99** | 1.8 | **99** | 6.1 | **99** | 1.2 |
| Exp3-2 | **85** | 6.5 | **85** | 7.4 | **85** | 0.6 | **85** | 0.3 | **85** | 6.4 | **85** | 2.5 |
| Exp3-3 | **86** | 4.0 | **86** | 3.4 | **86** | 0.6 | **86** | 0.1 | **86** | 9.7 | **86** | 2.2 |
| Exp3-4 | **111** | 19.0 | **111** | 67.4 | **111** | 7.4 | **111** | 1.3 | **111** | 9.5 | **111** | 2.3 |
| Exp4-1 | **112** | 196.2 | **112** | 900.9 | **112** | 198.2 | **112** | 11.9 | **112** | 294.0 | **112** | 74.6 |
| Exp4-2 | **87** | 46.9 | **87** | 456.2 | **87** | 21.6 | **87** | 2.5 | **87** | 91.7 | **87** | 23.1 |
| Exp4-3 | **89** | 48.6 | **89** | 510.4 | **89** | 6.9 | **89** | 3.3 | **89** | 112.1 | **89** | 27.0 |
| Exp4-4 | **121** | 158.1 | **121** | 759.6 | **121** | 77.9 | **121** | 3.8 | 126 | 326.6 | 126 | 96.1 |
| Exp5-1 | **87** | 10.8 | **87** | 12.6 | **87** | 2.3 | **87** | 1.2 | **87** | 0.7 | **87** | 0.4 |
| Exp5-2 | **69** | 1.7 | **69** | 9.5 | **69** | 0.4 | **69** | 0.3 | **69** | 0.4 | **69** | 0.2 |
| Exp5-3 | **74** | 5.0 | **74** | 19.5 | **74** | 1.8 | **74** | 0.8 | **74** | 0.6 | **74** | 0.3 |
| Exp5-4 | **96** | 9.2 | **96** | 8.3 | **96** | 3.5 | **96** | 1.4 | **96** | 0.7 | **96** | 0.4 |
| Exp6-1 | **118** | 77.7 | **118** | 506.0 | **118** | 65.7 | **118** | 5.1 | **118** | 113.6 | **118** | 12.3 |
| Exp6-2 | **98** | 8.7 | **98** | 9.0 | **98** | 4.7 | **98** | 1.2 | **98** | 67.7 | **98** | 4.1 |
| Exp6-3 | **103** | 8.6 | **103** | 11.8 | **103** | 2.7 | **103** | 0.9 | **103** | 148.5 | **103** | 17.1 |
| Exp6-4 | **120** | 111.5 | **120** | 330.6 | **120** | 36.7 | **120** | 4.4 | **120** | 38.8 | **120** | 5.7 |
| Exp7-1 | 117 | 906.2 | 116 | 900.8 | 117 | 900.8 | **111** | 586.4 | 113 | 900.0 | 112 | 900.0 |
| Exp7-2 | **79** | 335.0 | ***79*** | 900.2 | **79** | 249.2 | **79** | 1.8 | ***79*** | 900.0 | **79** | 97.8 |
| Exp7-3 | **83** | 544.1 | 85 | 900.3 | **83** | 214.2 | **83** | 4.0 | 87 | 900.0 | **83** | 76.6 |
| Exp7-4 | 131 | 907.1 | 135 | 900.8 | 130 | 901.4 | **126** | 199.0 | 129 | 900.0 | ***126*** | 900.0 |
| Exp8-1 | **161** | 8.2 | **161** | 11.7 | **161** | 4.7 | **161** | 0.1 | 161 | 900.0 | **161** | 0.0 |
| Exp8-2 | **151** | 5.7 | **151** | 12.0 | **151** | 3.0 | **151** | 0.1 | ***151*** | 900.0 | **151** | 0.0 |
| Exp8-3 | **153** | 6.1 | **153** | 4.7 | **153** | 3.1 | **153** | 0.1 | ***153*** | 900.0 | **153** | 0.0 |
| Exp8-4 | **163** | 12.8 | **163** | 31.7 | **163** | 5.5 | **163** | 0.2 | ***163*** | 900.0 | **163** | 0.7 |
| Exp9-1 | **116** | 7.0 | **116** | 7.0 | **116** | 5.2 | **116** | 1.2 | **116** | 8.7 | **116** | 0.5 |
| Exp9-2 | **102** | 3.4 | **102** | 6.8 | **102** | 1.7 | **102** | 0.4 | **102** | 6.3 | **102** | 0.4 |
| Exp9-3 | **105** | 4.5 | **105** | 4.6 | **105** | 1.9 | **105** | 0.4 | **105** | 7.9 | **105** | 0.6 |
| Exp9-4 | **120** | 4.3 | **120** | 26.9 | **120** | 3.8 | **120** | 1.6 | **120** | 7.0 | **120** | 0.8 |
| Exp10-1 | **146** | 129.7 | **146** | 875.5 | **146** | 20.2 | **146** | 2.2 | 153 | 900.0 | 147 | 618.1 |
| Exp10-2 | **135** | 71.7 | **135** | 86.1 | **135** | 12.0 | **135** | 1.2 | ***135*** | 900.0 | ***135*** | 900.0 |
| Exp10-3 | **137** | 38.1 | **137** | 104.2 | **137** | 34.7 | **137** | 1.3 | 137 | 900.0 | ***137*** | 900.0 |
| Exp10-4 | **157** | 364.6 | 160 | 900.6 | **157** | 121.6 | **157** | 18.3 | 159 | 900.0 | 158 | 442.0 |

Note: The optimal objective values are indicated in bold font, and those that are optimal but not verified/finalized are shown in italic bold font.

objective value obtained by each method within the time limit.

From Table 4, one can see that A-SB-B&B algorithm has better performance by the second lower-bounding strategy ($LB_2$) compared to the first lower bound ($LB_1$); hence, for the other two SB-B&B algorithms the second strategy is reported. One can also see that the A-SB-B&B algorithm is very efficient, while the M-SB-B&B and C-SB-B&B algorithms are not. However, one should note that the A-SB-B&B algorithm may find slightly suboptimal solutions; in fact, it stops at optimality except for six out of 40 instances, i.e. Exp3-1, Exp6-4, Exp9-1, Exp9-2, Exp9-4, and Exp10-4.

The CP model performs better than the MILP model, which cannot solve 14 instances in the 900-second time limit, while the CP model solves them all in shorter run times. The A-SB-B&B algorithm cannot beat the CP model, but it outperforms the MILP model in most instances (except instances Exp3-1, Exp9-1, Exp9-2, Exp9-4, Exp10-2, and Exp10-4).

To have a fair comparison, the ASO algorithm proposed by Ahmadi-Javid and Hooshangi-Tabrizi (2015; 2017), which outperforms the other heuristics, is adjusted for the heterogeneous case. The adjusted ASO algorithm were ran 10 times due its stochastic nature, and Table 5 reports the best result (this is in favour of this algorithm). However, one can see that the ASO algorithm works very poorly compared to the other methods and can find the optimal solution in only seven instances in 10 runs.

### 4.1. Numerical results for the homogenous case

This subsection evaluates the solution methods for the homogeneous case. Table 5 reports our numerical study on the instances used by Bilge and Ulusoy (1995). In this table, the old MILP is obtained by linearising the nonlinear formulation developed by Bilge and Ulusoy (1995) for the homogeneous case, and the new MILP model is the one presented in this paper (it can be used for the homogeneous case when the input speeds are set equally). The adjusted MILP model is the adjustment of the new MILP model for the homogeneous case, given below:

$$\min \quad CT^{Jo}_{max} \tag{30}$$

subject to:

$$CT^{Jo}_{max} \geq ST^{Op}_i + p_i \quad j \in J, i \in I : i = N_j + n_j \tag{31}$$

$$ST^{Op}_i - ST^{Op}_{i-1} \geq p_{i-1} + d_{m(i-1),m(i)} \qquad j \in J,\ i \in I_j : i \geq N_j + 2 \tag{32}$$

$$ST^{Op}_i \geq d_{L/U,m(i)} \quad j \in J,\ i \in I : i = N_j + 1 \tag{33}$$

$$ST^{Op}_i \geq ST^{Op}_h + p_h - H \times q_{i,h} \qquad i, h \in I : m(i) = m(h), i \neq h \tag{34}$$

$$q_{i,h} + q_{h,i} = 1 \quad i, h \in I : m(i) = m(h), i \neq h \tag{35}$$

$$x_{0,i,t} + \sum_{h \in I^{es}_i} x_{h,i} = 1 \quad i \in I \tag{36}$$

$$x_{h,0,t} + \sum_{i \in I^{ep}_h} x_{h,i} = 1 \quad i \in I \tag{37}$$

$$\sum_{i \in I} x_{0,i} \leq |T| \tag{38}$$

$$\sum_{i \in I} x_{0,i} - \sum_{h \in I} x_{h,0} = 0 \tag{39}$$

$$CT^{Tr}_i \leq ST^{Op}_i \quad i \in I \tag{40}$$

$$CT^{Tr}_i - d_{m(i-1),m(i)} \geq ST^{Op}_{i-1} + p_{i-1} \qquad j \in J, i \in I_j : i \geq N_j + 2 \tag{41}$$

$$CT^{Tr}_h \geq d_{L\backslash U,m(h)} + H \times (x_{0,h} - 1) \qquad t \in T, j \in J, h = N_j + 1 \tag{42}$$

$$CT^{Tr}_h \geq d_{L/U,m(h-1)} + d_{m(h-1),m(h)} + H \times (x_{0,h} - 1) \qquad t \in T, j \in J, h \in I_j, h \neq N_j + 1 \tag{43}$$

$$CT^{Tr}_h \geq CT^{Tr}_i + d_{m(i),L/U} + d_{L/U,m(h)} + H \times (x_{i,h} - 1) \qquad t \in T,\ i \in I,\ j \in J,\ h = N_j + 1,\ h \neq i \tag{44}$$

$$CT^{Tr}_h \geq CT^{Tr}_i + d_{m(i),m(h-1)} + d_{m(h-1),m(h)} + H \times (x_{i,h} - 1) \qquad t \in T,\ i \in I,\ j \in J,\ h \in I_j h \neq N_j + 1,\ h \neq i \tag{45}$$

$$ST^{Op}_i \geq 0, CT^{Tr}_i \geq 0, CT^{Jo}_{max} \geq 0 \quad i \in I \tag{46}$$

$$q_{i,h} \in \{0, 1\} \quad i, h \in I : m(i) = m(h) \tag{47}$$

$$x_{h,i} \in \{0, 1\} \quad i, h \in I \cup \{0\}, t \in T \tag{48}$$

where the new variables (obtained by removing the transporter index $t$) are defined as follows:

$x_{h,i}$: 1 if a transporter delivers the job from $m(pre_i)$ to $m(i)$ after delivering the job to machine $m(h)$, $i, h \in I$;

$x_{0,i}$: 1 if a transporter delivers the job from $m(pre_i)$ to $m(i)$ after being initialised from L\U, $i \in I,\ t \in T$;

$x_{h,0}$: 1 if a transporter performs the last unloaded trip to L\U after delivering the job to machine $m(h)$, $h \in I$.

From Table 5, one can see that all the methods terminated faster compared to the heterogeneous case, which is expected because solving the homogenous case is easier. The new MILP model performs successfully for the homogenous case, but it is slower than the old MILP, which is specially developed for the homogenous case. However, the adjusted new MILP is considerably faster than the old model. The A-SB-B&B algorithm with the second lower bound performs better than the first lower bound. The A-SB-B&B algorithm also finds optimal solutions faster than the new MILP model in most cases. It also performs much better on finding optimal solutions within 900 s in comparison with the heterogonous case. The CP model remains as the best solution method. These observations agree with the ones made for the heterogonous case.

## 5. Managerial implications

This section provides practical insights of our study. Firstly, it is shown how our results can be used to analyse the number and speeds of transporters. Secondly, the impact of having a fleet of heterogeneous transporters is evaluated in comparison with a corresponding homogeneous case. Thirdly, it is evaluated how much the scheduling plans deviate from the optimality if the differences among the transporters are ignored.

### 5.1. *A sensitivity analysis on properties of transportation fleet*

Deciding on the number and technology of transporters can be a key factor in a job-shop system to achieve a high performance. In this subsection, the sensitivity of the optimal makespan with respect to both the number of the transporters and their moving speed is analysed. To this end, instance Exp10-1 is considered. The number of transporters is set to $l = 1, \ldots, 5$ and speeds can be any of 1, 1.5, 2, 2.5, 3, 3.5, 4.

Figure 2 distinguishes the areas with approximately equal values of the makespan where the makespan reduces by moving from the red area to the blue area. Based on this figure, for example, it can be seen that for two transporters with the speed 2 one obtains smaller values for the makespan rather than the case with three

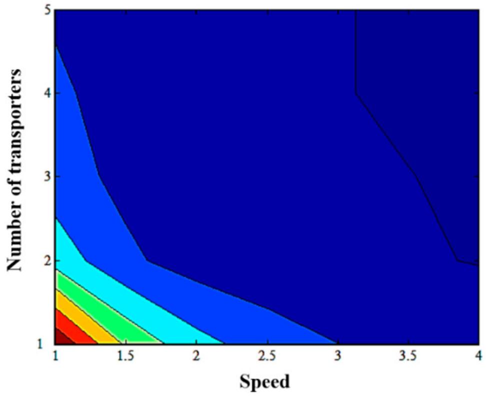



**Figure 2.** An illustration of how the optimal makespan varies by the number and speed of transporters.

**Table 6.** A comparison of a fleet of heterogeneous transporters (instances 2–5) with the corresponding fleet of homogenous transporters (instance 1).

| Instance | v1 | v2 | Makespan | Run time (s) | # of assignments to first transporter | # of assignments to second transporter |
|---|---|---|---|---|---|---|
| 1 | 0.5 | 0.5 | 230.0 | 1344 | 10 | 11 |
| 2 | 0.4 | 0.6 | 224.3 | 568 | 8 | 13 |
| 3 | 0.3 | 0.7 | 226.4 | 768 | 7 | 14 |
| 4 | 0.2 | 0.8 | 224.0 | 268 | 5 | 16 |
| 5 | 0.1 | 0.9 | 224.1 | 103 | 2 | 19 |

transporters with the speed 1. Also, increasing the number or speeds of the transporters from smaller values results in more improvement of the makespan. For example, changing the speed from 1 to 1.5 for all the transporters yields more improvement than changing the speed from 3 to 3.5. In fact, any increase beyond these in the number or speeds of the transporters does make a significant improvement. A similar study can be done for analysing a fleet of heterogeneous transporters, but a visual illustration (similar to Figure 2) cannot be depicted.

### 5.2. *Advantage of a differentiated transportation fleet*

The question addressed here is whether a differentiated transportation fleet can be better than a fleet of the same transporters or not. To answer this question, for each fleet of heterogeneous transporters, the corresponding fleet of homogenous transporters is defined such that the speed of each transporter is set to the average of the heterogeneous transporters' speeds (the sums of the speeds in both cases are equal).

In Table 6, five instances are generated based on instance Exp10-1. The instances 2–5 have a fleet of heterogeneous transporters, while the instance 1 has the corresponding fleet of homogeneous transporters (in each instance the sum of the speeds is equal to 1). From this table, one can see that the makespan value for the homogenous case is always greater than those obtained in the heterogeneous case. Another observation is that the run times in the heterogeneous case are less than those required for the homogenous case. Moreover, as intuitively expected, the number of assignments to a transporter with higher speed is larger.

### 5.3. *Scheduling Inferiority caused by ignoring the transporter differences*

Consider the case that the transporters have different speeds in reality, but in the scheduling phase one simplifies the case by assuming that all of them have the same speed (the average of the real speeds), which is commonly assumed in the previous studies. Let us evaluate how much this simplification results in inferior scheduling schemes. To see what happens, it is sufficient to measure the deviation of the makespan of the scheduling obtained after the simplification (where the speeds are set equally) from the makespan of the scheduling obtained by considering the real speeds without any approximation. In Figure 3, for instance Exp10-1, the deviations from optimality are depicted in percentages. One can see that the deviation reaches about 400% when the speed of a transporter is nine times of the speed of the other. This clearly clarifies the practical importance of the problem studied here.

## 6. Conclusions

The previous studies on integrated scheduling of machines and transporters assumed that the transporters are homogenous. This paper for the first time studies a job-shop scheduling problem with a limited number of heterogeneous transporters. The problem is first formulated as MILP and CP models, which can be solved by LP-based B&B and direct search methods, respectively. Then, several exact and heuristic structure-based B&B algorithms with two bounding procedures are developed. These solution methods, along with the best heuristic adjusted from the literature, are compared and analysed on a set of benchmark test problems.

The developed models enable us to carry out a trade-off analysis between the number of transporters and their speeds. It is observed that having fewer but faster transporters can have better performance than having more and slower transporters. In fact, the number and speeds of the transporters in a job shop can be set appropriately using our results if a suitable sensitivity analysis is carried out. It is also observed that a differentiated fleet of transporters can work far better in a job-shop

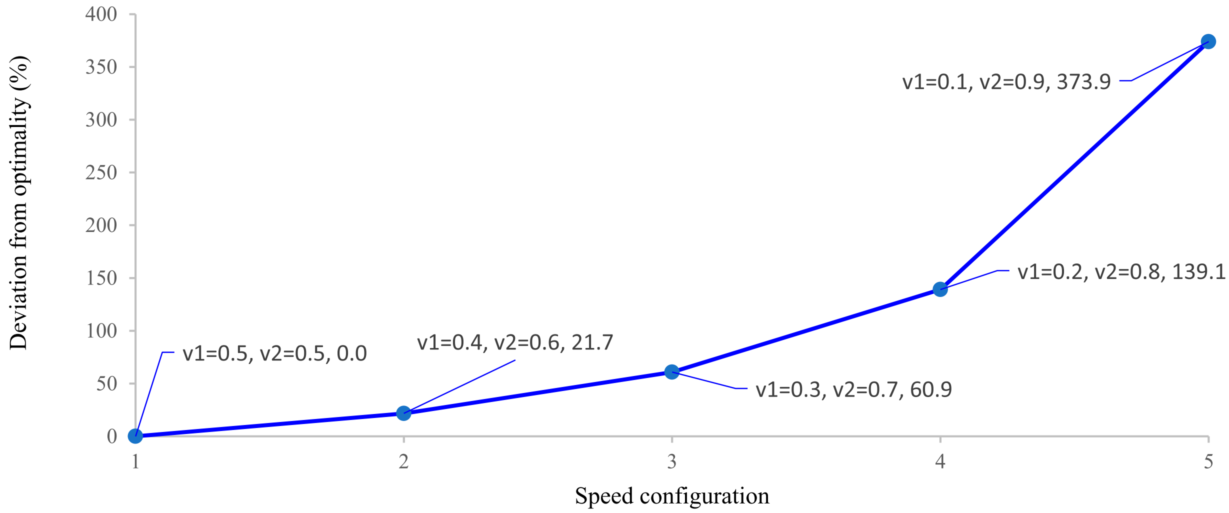



**Figure 3.** The impact of ignoring the differences of the transporters' speeds.

environment compared to the case of a fleet of homogeneous transporters whose speeds are equally set to the average speed in the differentiated feet. This may be an interesting observation which suggests that one can improve the makespan by having a combination of slow and fast transporters, at the same level of cost (energy usage). It is also shown that ignoring the differences of the transporters yields highly inferior scheduling schemes, which highlights the importance of studying the problem considered in this paper.

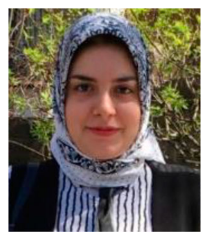
Future studies can develop other exact or heuristic algorithms or study new problems extending our assumptions. For example, other performance metrics instead of makespan, random failures of the machines or transporters, or a dynamic stochastic environment can be considered. Another area of future research is studying an integrated scheduling problem in which the speeds can be set optimally for different transportation tasks such that the level of energy usage is controlled.

## Disclosure statement

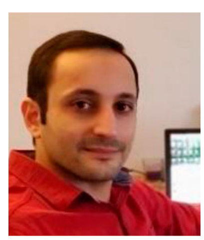
No potential conflict of interest was reported by the author(s).

## Notes on contributors

*Amir Ahmadi-Javid* is an Associate Professor of Operations Management and Business Analytics at Amirkabir University of Technology (Tehran Polytechnic). He has published numerous research articles on different planning aspects of supply chains and service systems, risk analysis, financial engineering, project management, probability and statistics, and optimisation. He is a professional expert in developing efficient management solutions for different industries and business areas.



*Maryam Haghi* is a researcher in the field of Operations Research and Data Analysis, holding a Ph.D. in Industrial Engineering from Concordia University and a postdoctoral experience from HEC Montréal. Her expertise spans operations research, stochastic programming, mixed-integer linear programming, decomposition-based algorithms, and heuristics. Her research outcomes have been used for optimising various systems and processes in healthcare, logistics, and transportation, driving efficiency and performance improvements in these industries.



*Pedram Hooshangi-Tabrizi* is a Principal Operations Research Engineer with a Ph.D. in Industrial Engineering from Concordia University. His research interests lie in the areas of integer programming, stochastic programming, robust optimisation, and linear programming. He has applied his expertise in mathematical modelling, optimisation techniques, and advanced methodologies to various industries, including scheduling problems, network optimisation, healthcare, inventory management, and production planning.

## Data availability statement

Derived data supporting the findings of this study are available from the corresponding author Dr. Amir Ahmadi-Javid on request.

## ORCID

*Amir Ahmadi-Javid* http://orcid.org/0000-0002-9132-6887

## References

Abdelmaguid, T. F., A. O. Nassef, B. A. Kamal, and M. F. Hassan. 2004. "A Hybrid GA/Heuristic Approach to the Simultaneous Scheduling of Machines and Automated Guided Vehicles." *International Journal of Production*

*Research* 42: 267–281. https://doi.org/10.1080/0020754032000123579.

Ahmadi-Javid, A., and P. Hooshangi-Tabrizi. 2015. "A Mathematical Formulation and Anarchic Society Optimisation Algorithms for Integrated Scheduling of Processing and Transportation Operations in a Flow-Shop Environment." *International Journal of Production Research* 53: 5988–6006. https://doi.org/10.1080/00207543.2015.1035812.

Ahmadi-Javid, A., and P. Hooshangi-Tabrizi. 2017. "Integrating Employee Timetabling with Scheduling of Machines and Transporters in a Job-shop Environment: A Mathematical Formulation and an Anarchic Society Optimization Algorithm." *Computers & Operations Research* 84: 73–91. https://doi.org/10.1016/j.cor.2016.11.017.

Anwar, M. F., and R. Nagi. 1998. "Integrated Scheduling of Material Handling and Manufacturing Activities for Just-in-Time Production of Complex Assemblies." *International Journal of Production Research* 36: 653–681. https://doi.org/10.1080/002075498193624.

Baldacci, R., M. Battarra, and D. Vigo. 2008. "Routing a Heterogeneous Fleet of Vehicles." In *The Vehicle Routing Problem: Latest Advances and new Challenges*, edited by B. Golden, S. Raghavan, and E. Wasil, 3–27. Boston, MA: Springer.

Baruwa, O. T., and M. A. Piera. 2016. "A Coloured Petri net-Based Hybrid Heuristic Search Approach to Simultaneous Scheduling of Machines and Automated Guided Vehicles." *International Journal of Production Research* 54 (16): 4773–4792. https://doi.org/10.1080/00207543.2015.1087656.

Bilge, Ü, and G. Ulusoy. 1995. "A Time Window Approach to Simultaneous Scheduling of Machines and Material Handling System in an FMS." *Operations Research* 43: 1058–1070. https://doi.org/10.1287/opre.43.6.1058.

Boyer, V., J. Vallikavungal, X. C. Rodríguez, and M. A. Salazar-Aguilar. 2021. "The Generalized Flexible Job Shop Scheduling Problem." *Computers & Industrial Engineering* 160: 107542. https://doi.org/10.1016/j.cie.2021.107542.

Brucker, P., B. Jurisch, and B. Sievers. 1994. "A Branch and Bound Algorithm for the Job-shop Scheduling Problem." *Discrete Applied Mathematics* 49: 107–127. https://doi.org/10.1016/0166-218X(94)90204-6.

Bürgy, R., and H. Gröflin. 2016. "The Blocking Job Shop with Rail-bound Transportation." *Journal of Combinatorial Optimization* 31 (1): 152–181. https://doi.org/10.1007/s10878-014-9723-3.

Buzacott, J. A., and D. D. Yao. 1986. "Flexible Manufacturing Systems: A Review of Analytical Models." *Management Science* 32 (7): 890–905. https://doi.org/10.1287/mnsc.32.7.890.

Cai, L., W. Li, Y. Luo, and L. He. 2022. "Real-time Scheduling Simulation Optimisation of Job Shop in a Production-logistics Collaborative Environment." *International Journal of Production Research*, https://doi.org/10.1080/00207543.2021.2023777.

Caumond, A., P. Lacomme, A. Moukrim, and N. Tchernev. 2009. "An MILP for Scheduling Problems in an FMS with one Vehicle." *European Journal of Operational Research* 199: 706–722. https://doi.org/10.1016/j.ejor.2008.03.051.

Chaudhry, I. A., S. Mahmood, and M. Shami. 2011. "Simultaneous Scheduling of Machines and Automated Guided Vehicles in Flexible Manufacturing Systems Using Genetic Algorithms." *Journal of Central South University* 18: 1473–1486. https://doi.org/10.1007/s11771-011-0863-7.

Chen, J.-S. 2006. "A Branch and Bound Procedure for the Reentrant Permutation Flow-shop Scheduling Problem." *The International Journal of Advanced Manufacturing Technology* 29 (11): 1186–1193. https://doi.org/10.1007/s00170-005-0017-x.

Chung, C.-S., J. Flynn, and Ö Kirca. 2006. "A Branch and Bound Algorithm to Minimize the Total Tardiness for m-Machine Permutation Flowshop Problems." *European Journal of Operational Research* 174: 1–10. https://doi.org/10.1016/j.ejor.2004.12.023.

Dang, Q.-V., C. T. Nguyen, and H. Rudová. 2019. "Scheduling of Mobile Robots for Transportation and Manufacturing Tasks." *Journal of Heuristics* 25 (2): 175–213. https://doi.org/10.1007/s10732-018-9391-z.

Dell'Amico, M. 1996. "Shop Problems with Two Machines and Time Lags." *Operations Research* 44 (5): 777–787. https://doi.org/10.1287/opre.44.5.777.

Deroussi, L., M. Gourgand, and N. Tchernev. 2008. "A Simple Metaheuristic Approach to the Simultaneous Scheduling of Machines and Automated Guided Vehicles." *International Journal of Production Research* 46: 2143–2164. https://doi.org/10.1080/00207540600818286.

ElMaraghy, H. A. 2005. "Flexible and Reconfigurable Manufacturing Systems Paradigms." *International Journal of Flexible Manufacturing Systems* 17 (4): 261–276. https://doi.org/10.1007/s10696-006-9028-7.

Fondrevelle, J., A. Oulamara, and M. C. Portmann. 2006. "Permutation Flowshop Scheduling Problems with Maximal and Minimal Time Lags." *Computers & Operations Research* 33 (6): 1540–1556. https://doi.org/10.1016/j.cor.2004.11.006.

Fontes, D. B. M., and S. M. Homayouni. 2019. "Joint Production and Transportation Scheduling in Flexible Manufacturing Systems." *Journal of Global Optimization* 74 (4): 879–908. https://doi.org/10.1007/s10898-018-0681-7.

Gao, K., Z. Cao, L. Zhang, Z. Chen, Y. Han, and Q. Pan. 2019. "A Review on Swarm Intelligence and Evolutionary Algorithms for Solving Flexible Job Shop Scheduling Problems." *IEEE/CAA Journal of Automatica Sinica* 6 (4): 904–916. https://doi.org/10.1109/JAS.2019.1911540.

Graham, R. L., E. L. Lawler, J. K. Lenstra, and A. H. G. R. Kan. 1979. "Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey." *Annals of Discrete Mathematics* 5: 287–326. https://doi.org/10.1016/S0167-5060(08)70356-X.

Gultekin, H., B. Coban, and V. E. Akhlaghi. 2018. "Cyclic Scheduling of Parts and Robot Moves in m-Machine Robotic Cells." *Computers & Operations Research* 90: 161–172. https://doi.org/10.1016/j.cor.2017.09.018.

Ham, A. 2021. "Transfer-robot Task Scheduling in Job Shop." *International Journal of Production Research* 59 (3): 813–823. https://doi.org/10.1080/00207543.2019.1709671.

Hamdi, I., and T. Loukil. 2015. "Upper and Lower Bounds for the Permutation Flowshop Scheduling Problem with Minimal Time Lags." *Optimization Letters* 9 (3): 465–482. https://doi.org/10.1007/s11590-014-0761-7.

Harrabi, M., O. B. Driss, and K. Ghedira. 2021. "A Hybrid Evolutionary Approach to Job-shop Scheduling with Generic Time Lags." *Journal of Scheduling* 24 (3): 329–346. https://doi.org/10.1007/s10951-021-00683-w.

Homayouni, S. M., and D. B. Fontes. 2021. "Production and Transport Scheduling in Flexible Job Shop Manufacturing

Systems." *Journal of Global Optimization* 79 (2): 463–502. https://doi.org/10.1007/s10898-021-00992-6.

Hu, L., Z. Liu, W. Hu, Y. Wang, J. Tan, and F. Wu. 2020. "Petri-net-based Dynamic Scheduling of Flexible Manufacturing System via Deep Reinforcement Learning with Graph Convolutional Network." *Journal of Manufacturing Systems* 55: 1–14. https://doi.org/10.1016/j.jmsy.2020.02.004.

Hurink, J., and S. Knust. 2001. "Makespan Minimization for Flow-Shop Problems with Transportation Times and a Single Robot." *Discrete Applied Mathematics* 112: 199–216. https://doi.org/10.1016/S0166-218X(00)00316-4.

Hurink, J., and S. Knust. 2002. "A Tabu Search Algorithm for Scheduling a Single Robot in a Job-shop Environment." *Discrete Applied Mathematics* 119: 181–203. https://doi.org/10.1016/S0166-218X(01)00273-6.

Hurink, J., and S. Knust. 2005. "Tabu Search Algorithms for Job-shop Problems with a Single Transport Robot." *European Journal of Operational Research* 162: 99–111. https://doi.org/10.1016/j.ejor.2003.10.034.

Jerald, J., P. Asokan, R. Saravanan, and A. D. C. Rani. 2006. "Simultaneous Scheduling of Parts and Automated Guided Vehicles in an FMS Environment Using Adaptive Genetic Algorithm." *The International Journal of Advanced Manufacturing Technology* 29: 584–589. https://doi.org/10.1007/s00170-005-2529-9.

Khayat, G. E., A. Langevin, and D. Riopel. 2006. "Integrated Production and Material Handling Scheduling Using Mathematical Programming and Constraint Programming." *European Journal of Operational Research* 175: 1818–1832. https://doi.org/10.1016/j.ejor.2005.02.077.

Koç, Ç, T. Bektaş, O. Jabali, and G. Laporte. 2016. "Thirty Years of Heterogeneous Vehicle Routing." *European Journal of Operational Research* 249 (1): 1–21. https://doi.org/10.1016/j.ejor.2015.07.020.

Kumar, M. S., R. Janardhana, and C. S. P. Rao. 2011. "Simultaneous Scheduling of Machines and Vehicles in an FMS Environment with Alternative Routing." *The International Journal of Advanced Manufacturing Technology* 53: 339–351. https://doi.org/10.1007/s00170-010-2820-2.

Lacomme, P., M. Larabi, and N. Tchernev. 2013. "Job-shop Based Framework for Simultaneous Scheduling of Machines and Automated Guided Vehicles." *International Journal of Production Economics* 143: 24–34. https://doi.org/10.1016/j.ijpe.2010.07.012.

Lacomme, P., A. Moukrim, and N. Tchernev. 2005. "Simultaneous Job Input Sequencing and Vehicle Dispatching in a Single-vehicle Automated Guided Vehicle System: A Heuristic Branch-and-Bound Approach Coupled with a Discrete Events Simulation Model." *International Journal of Production Research* 43: 1911–1942. https://doi.org/10.1080/13528160412331326450.

Langston, M. A. 1987. "Interstage Transportation Planning in the Deterministic Flow-shop Environment." *Operations Research* 35: 556–564. https://doi.org/10.1287/opre.35.4.556.

Li, J. Q., Y. Du, K. Z. Gao, P. Y. Duan, D. W. Gong, Q. K. Pan, and P. N. Suganthan. 2021. "A Hybrid Iterated Greedy Algorithm for a Crane Transportation Flexible Job Shop Problem." *IEEE Transactions on Automation Science and Engineering*, https://doi.org/10.1109/TASE.2021.3062979.

Li, Y., W. Gu, M. Yuan, and Y. Tang. 2022. "Real-time Data-Driven Dynamic Scheduling for Flexible Job Shop with Insufficient Transportation Resources Using Hybrid Deep Q Network." *Robotics and Computer-Integrated Manufacturing* 74: 102283. https://doi.org/10.1016/j.rcim.2021.102283.

Liang, Z., M. Liu, P. Zhong, and C. Zhang. 2021. "Application Research of a New Neighbourhood Structure with Adaptive Genetic Algorithm for Job Shop Scheduling Problem." *International Journal of Production Research*, https://doi.org/10.1080/00207543.2021.2007310.

Lin, J. T., C. C. Chiu, and Y. H. Chang. 2019. "Simulation-based Optimization Approach for Simultaneous Scheduling of Vehicles and Machines with Processing Time Uncertainty in FMS." *Flexible Services and Manufacturing Journal* 31 (1): 104–141. https://doi.org/10.1007/s10696-017-9302-x.

Liu, S. Q., and E. Kozan. 2017. "A Hybrid Metaheuristic Algorithm to Optimise a Real-world Robotic Cell." *Computers & Operations Research* 84: 188–194. https://doi.org/10.1016/j.cor.2016.09.011.

Márquez, C. R., and C. C. Ribeiro. 2022. "Shop Scheduling in Manufacturing Environments: A Review." *International Transactions in Operational Research* 29 (6): 3237–3293. https://doi.org/10.1111/itor.13108.

Martinez-Barbera, H., and D. Herrero-Perez. 2010. "Development of a Flexible AGV for Flexible Manufacturing Systems." *Industrial Robot: An International Journal* 37 (5): 459–468. https://doi.org/10.1108/01439911011063281.

Naderi, B., A. A. Ahmadi-Javid, and F. Jolai. 2010. "Permutation Flowshops with Transportation Times: Mathematical Models and Solution Methods." *The International Journal of Advanced Manufacturing Technology* 46: 631–647. https://doi.org/10.1007/s00170-009-2122-8.

Nouri, H. E., O. B. Driss, and K. Ghédira. 2016a. "A Classification Schema for the Job Shop Scheduling Problem with Transportation Resources: State-of-the-art Review." In *Artificial Intelligence Perspectives in Intelligent Systems. Advances in Intelligent Systems and Computing*, Vol. 464, edited by R. Silhavy, R. Senkerik, Z. Oplatkova, P. Silhavy, and Z. Prokopova, 1–11. Cham: Springer.

Nouri, H. E., O. B. Driss, and K. Ghédira. 2016b. "Simultaneous Scheduling of Machines and Transport Robots in Flexible Job Shop Environment Using Hybrid Metaheuristics Based on Clustered Holonic Multiagent Model." *Computers & Industrial Engineering* 102: 488–501. https://doi.org/10.1016/j.cie.2016.02.024.

Park, K. T., S. W. Jeon, and S. D. Noh. 2022. "Digital Twin Application with Horizontal Coordination for Reinforcement-learning-based Production Control in a Re-entrant Job Shop." *International Journal of Production Research* 60 (7): 2151–2167. https://doi.org/10.1080/00207543.2021.1884309.

Parker, L. E., D. Rus, and G. S. Sukhatme. 2016. "Multiple Mobile Robot Systems." In *Springer Handbook of Robotics*, edited by B. Siciliano, and O. Khatib, 1335–1384. Cham: Springer.

Petrović, M., Z. Miljković, and A. Jokić. 2019. "A Novel Methodology for Optimal Single Mobile Robot Scheduling Using Whale Optimization Algorithm." *Applied Soft Computing* 81: 105520. https://doi.org/10.1016/j.asoc.2019.105520.

Pinedo, M. 2005. *Planning and Scheduling in Manufacturing and Services*. New York: Springer.

Raman, N. 1986. "Simultaneous Scheduling of Machines and Material Handling Devices in Automated Manufacturing."

In *Proceeding of the Second ORSA/TIMS Conference on Flexible Manufacturing Systems: Operations Research Models and Applications*, edited by K. E. Stecke, and R. Suri, 455–465. Amsterdam: Elsevier.

Rebaine, D., and V. A. Strusevich. 1999. "Two-machine Open Shop Scheduling with Special Transportation Times." *Journal of the Operational Research Society* 50: 756–764. https://doi.org/10.1057/palgrave.jors.2600769.

Reddy, B. S. P., and C. S. P. Rao. 2006. "A Hybrid Multi-objective GA for Simultaneous Scheduling of Machines and AGVs in FMS." *The International Journal of Advanced Manufacturing Technology* 31: 602–613. https://doi.org/10.1007/s00170-005-0223-6.

Ren, W., Y. Yan, Y. Hu, and Y. Guan. 2021. "Joint Optimisation for Dynamic Flexible Job-shop Scheduling Problem with Transportation Time and Resource Constraints." *International Journal of Production Research*, https://doi.org/10.1080/00207543.2021.1968526.

Rossit, D. A., F. Tohmé, and M. Frutos. 2018. "The Non-permutation Flow-shop Scheduling Problem: A Literature Review." *Omega* 77: 143–153. https://doi.org/10.1016/j.omega.2017.05.010.

Sabuncuoglu, I., and S. Karabuk. 1998. "A Beam Search-based Algorithm and Evaluation of Scheduling Approaches for Flexible Manufacturing Systems." *IIE Transactions* 30: 179–191. https://doi.org/10.1080/07408179808966449.

Saidi-Mehrabad, M., S. Dehnavi-Arani, F. Evazabadian, and V. Mahmoodian. 2015. "An Ant Colony Algorithm (ACA) for Solving the New Integrated Model of Job Shop Scheduling and Conflict-free Routing of AGVs." *Computers & Industrial Engineering* 86: 2–13. https://doi.org/10.1016/j.cie.2015.01.003.

Samarghandi, H. 2019. "Minimizing the Makespan in a Flow Shop Environment Under Minimum and Maximum Time-lag Constraints." *Computers & Industrial Engineering* 136: 614–634. https://doi.org/10.1016/j.cie.2019.07.048.

Schulz, S., M. Schönheit, and J. S. Neufeld. 2022. "Multi-objective Carbon-efficient Scheduling in Distributed Permutation Flow Shops Under Consideration of Transportation Efforts." *Journal of Cleaner Production* 365: 132551. https://doi.org/10.1016/j.jclepro.2022.132551.

Shady, S., T. Kaihara, N. Fujii, and D. Kokuryo. 2022. "A Novel Feature Selection for Evolving Compact Dispatching Rules Using Genetic Programming for Dynamic Job Shop Scheduling." *International Journal of Production Research*, https://doi.org/10.1080/00207543.2022.2053603.

Sun, Y., S. H. Chung, X. Wen, and H. L. Ma. 2021. "Novel Robotic Job-shop Scheduling Models with Deadlock and Robot Movement Considerations." *Transportation Research Part E: Logistics and Transportation Review* 149: 102273. https://doi.org/10.1016/j.tre.2021.102273.

Ulusoy, G., F. Sivrikaya-ŞerifoÇğlu, and Ü Bilge. 1997. "A Genetic Algorithm Approach to the Simultaneous Scheduling of Machines and Automated Guided Vehicles." *Computers & Operations Research* 24: 335–351. https://doi.org/10.1016/S0305-0548(96)00061-5.

Umar, U. A., M. K. A. Ariffin, N. Ismail, and S. H. Tang. 2015. "Hybrid Multiobjective Genetic Algorithms for Integrated Dynamic Scheduling and Routing of Jobs and Automated-guided Vehicle (AGV) in Flexible Manufacturing Systems (FMS) Environment." *The International Journal of Advanced Manufacturing Technology* 81 (9): 2123–2141. https://doi.org/10.1007/s00170-015-7329-2.

Verma, J. K., and V. Ranga. 2021. "Multi-robot Coordination Analysis, Taxonomy, Challenges and Future Scope." *Journal of Intelligent & Robotic Systems* 102 (1): 1–36. https://doi.org/10.1007/s10846-021-01378-2.

Xie, C., and T. Allen. 2015. "Simulation and Experimental Design Methods for Job Shop Scheduling with Material Handling: A Survey." *The International Journal of Advanced Manufacturing Technology* 80: 233–243. https://doi.org/10.1007/s00170-015-6981-x.

Xie, J., X. Li, L. Gao, and L. Gui. 2022. "A New Neighbourhood Structure for Job Shop Scheduling Problems." *International Journal of Production Research*. https://doi.org/10.1080/00207543.2022.2060772.

Xiong, H., S. Shi, D. Ren, and J. Hu. 2022. "A Survey of Job Shop Scheduling Problem: The Types and Models." *Computers & Operations Research* 142: 105731. https://doi.org/10.1016/j.cor.2022.105731.

Yadav, A., and S. C. Jayswal. 2018. "Modelling of Flexible Manufacturing System: A Review." *International Journal of Production Research* 56 (7): 2464–2487. https://doi.org/10.1080/00207543.2017.1387302.

Yu, W., H. Hoogeveen, and J. K. Lenstra. 2004. "Minimizing Makespan in a Two-machine Flow Shop with Delays and Unit-time Operations is NP-Hard." *Journal of Scheduling* 7 (5): 333–348. https://doi.org/10.1023/B:JOSH.0000036858.59787.c2.

Zhang, Q., H. Manier, and M. A. Manier. 2014. "A Modified Shifting Bottleneck Heuristic and Disjunctive Graph for job Shop Scheduling Problems with Transportation Constraints." *International Journal of Production Research* 52 (4): 985–1002. https://doi.org/10.1080/00207543.2013.828164.