Journal Pre-proof

Sustainable distributed permutation flow-shop scheduling model based on a triple bottom line concept

Amir M. Fathollahi-Fard , Lyne Woodward , Ouassima Akhrif

PII: S2452-414X(21)00033-9
DOI: https://doi.org/10.1016/j.jii.2021.100233
Reference: JII 100233

To appear in: *Journal of Industrial Information Integration*

Received date: 1 February 2021
Revised date: 6 April 2021
Accepted date: 9 June 2021

Please cite this article as: Amir M. Fathollahi-Fard , Lyne Woodward , Ouassima Akhrif , Sustainable distributed permutation flow-shop scheduling model based on a triple bottom line concept, *Journal of Industrial Information Integration* (2021), doi: https://doi.org/10.1016/j.jii.2021.100233

Authors' accepted manuscript. Article published in *Journal of Industrial Information Integration*, Volume 24, December 2021, 100233
https://doi.org/10.1016/j.jii.2021.100233

Highlights:

- A sustainable distributed permutation flow-shop scheduling model is developed.
- A novel multi-objective mixed integer linear programming approach is proposed.
- Different technologies, job opportunities and lost working days are considered.
- A novel heuristic based on the social engineering optimizer is introduced.
- Applicability of the approach is proven using an example from wood industry.

Authors' accepted manuscript. Article published in *Journal of Industrial Information Integration*, Volume 24, December 2021, 100233
https://doi.org/10.1016/j.jii.2021.100233

Sustainable distributed permutation flow-shop scheduling model based on a triple bottom line concept

Amir M. Fathollahi-Fard, Lyne Woodward*, Ouassima Akhrif

Department of Electrical Engineering, École de Technologie Supérieure, University of Québec, 1100, Notre-Dame St. W., Montréal, Canada

Sustainable distributed permutation flow shop scheduling model based on a triple bottom line concept

Abstract

Based on a triple bottom line concept, sustainable development is characterized by the simultaneous pursuit of economic, environmental and social goals. The implementation of this concept in production scheduling can result in the resolution of a sustainable Distributed Permutation Flow Shop Scheduling Problem (DPFSP). The present study conceptually shifts an energy-efficient DPFSP to a sustainable DPFSP, simultaneously contributing to economic, environmental and social improvements. The study aims not only to minimize the total energy consumption related to production, but also, to maximize, for the first time, the social factors linked to job opportunities and lost working days. Different production centers and technologies are considered as new suppositions to establish a sustainable DPFSP. In this regard, a novel multi-objective mixed integer linear model is developed. To manage the high complexity of the proposed model, a novel multi-objective learning-based heuristic is established, as an extension of the Social Engineering Optimizer (SEO). The applicability of the proposed model is determined in the context of the wood industry in Canada. Several simulated tests are considered to verify the model. The proposed heuristic is compared with one of the other well-known, recent and state-of-the art methods. In order to guarantee a fair comparison, the Taguchi method is used to tune the parameters of the algorithms. Finally, sensitivity analyses are done to assess the efficiency of the proposed model.

Keywords: Triple bottom line approach, production scheduling, distributed permutation flow shop scheduling problem, learning-based heuristic, social engineering optimizer.

1. Introduction

* Corresponding author, Email : lyne.woodward@etsmtl.ca

Authors' accepted manuscript. Article published in *Journal of Industrial Information Integration*, Volume 24, December 2021, 100233
https://doi.org/10.1016/j.jii.2021.100233

With sustainable development and social responsibility trending as a means of tackling environmental deterioration and other economic issues, the Triple Bottom Line (TBL) approach has become an active research topic in the supply chain, logistics and production management fields [1]. A sustainable production system is academically defined as a production system that takes economic, environmental and social factors into account [2]. Merging the concept of TBL with production systems opens up several new avenues for researches in terms of developing optimization models and algorithms for production planning [3-5]. In this context, the present work aims to find a way to model a sustainable production system for a wood processing company in Canada.

According to the government of Canada[1], production by the wood industry contributed a total of $19.8 billion to the country's GDP in 2013[2]. Hence, Canada has the largest market share of the wood industry in the world. However, based on the sustainable development paradigm and the TBL criteria, the Canadian wood industry's production systems must be redesigned to holistically include economic, environmental and social factors.

Many companies operating in the Canadian wood industry usually focus on economic performance, while ignoring social and environmental issues. Environmental sustainability is crucial as reports[1] indicate that in Canada, this industry is responsible for half of all carbon emissions. Machines in production centers use non-renewable energy, which represents a challenge in terms of meeting cleaner production goals.

Human life quality and social sustainability are intricately linked [1]. Indeed, in the literature, job opportunities and lost working days are two of the main factors that must be considered in order to achieve social sustainability meeting the guidelines of ISO 26000 [1, 6]. Each machine that uses a specific technology needs operators to run it. When the technology underlying such machines is updated, production systems are faced with lost work days, as the days are dedicated to teaching and updating operators' knowledge on this new technology. To address these issues, this study formulates an advanced multi-objective optimization model as a sustainable Distributed Permutation Flow Shop Scheduling Problem (DPFSP).

The literature on the DPFSP is very rich, and provides many optimization models and algorithms [4-5]. Obviously, the solution complexity of the DPFSP is very high, and is classified as NP-hard. Therefore, many heuristics have been developed to deal with this [7-12]. The multi-objective optimization model proposed in the present work is even more complex than what is seen in most current studies [12-14], as it integrates social and environmental factors to the DPFSP, as well as economic factors such as the makespan and the total cost. Real-life constraints of the industry such as multiple production centers and technology selection are also included. To the best of our knowledge, no existing optimization algorithm is suitable for solving our complex model due to the theory of no free lunch [15]. Based on this challenge, this study innovates a new single solution heuristic as a variation of the Social Engineering Optimizer (SEO) [16], using local search heuristics and a learning-based operator.

All in all, the present study makes the following contributions to the literature:

[1] https://www.nrcan.gc.ca/home
[2] Gross Domestic Product (GDP)

- A sustainable DPFSP based on the TBL concept is formulated as a new multi-objective mixed integer linear programming model;
- A novel learning-based SEO is heuristically introduced for solving the proposed problem;
- An industrial example of the wood industry in Canada is proposed to show the applicability of the simulation results.

The rest of the paper is broken down as follows: Section 2 evaluates recent and relevant studies in the area of DPFSP, with an identification of research gaps. Section 3 defines the sustainable DPFSP and establishes the formulation of this problem. Section 4 develops the solution representation of the optimization model, our new heuristic procedures and the proposed learning-based SEO. Section 5 introduces the industrial example of our model and provides simulation tests allowing an extensive analysis of the model and solutions developed. Section 6 presents the conclusion and recommendations for managers of production systems. It then goes on to present our results and, present future research directions.

2. Literature review

The DPFSP is a type of distributed production system in which production tasks are first assigned to different production centers, following which the system scheduling is planned and executed [7-8]. The DPFSP is academically an extension of the Permutation Flow Shop Scheduling Problem (PFSP), in which tasks are assigned among multiple production centers [9]. However, the general flow shop scheduling model schedules tasks only for one center [10]. Although many studies have applied the DPFSP to many industrial applications, such as automobile production and petrochemicals, most related studies have only looked at economic factors such as the makespan and tardiness, while ignoring sustainability criteria including energy-consumption and social benefits.

This section comprises a review of the most relevant works that have dealt with the DPFSP during the last decade. In 2010, Naderi and Ruiz [17] were the first to study the DPFSP. They solved a Mixed Integer Linear Programming (MILP) model with a view to reducing the makespan, using two heuristics for the assignment of designed production centers. Then, in 2011, Gao and Chen [18] proposed a Genetic Algorithm with Local Search Strategies (GALS) to address the DPFSP, taking the makespan into account. In 2013, Lin et al. [19] studied the DPFSP using a modified iterated greedy search algorithm. Similarly, in 2014, Naderi and Ruiz [20] developed a novel scatter search heuristic for this problem and compared its efficiency to that of other existing methods. In 2017, Bargaoui et al. [21] applied an optimization algorithm inspired by chemical reactions to address the DPFSP with the makespan criterion.

Two economic factors used in the literature are the makespan criterion which computes the maximum time of completion between all production centers, and the total flow-time criterion, which is the summation of the completion time for all these centers. In 2018, consideration of the total flow time criterion as an objective function was first proposed by Fernandez-Viagas et al. [22]. Then, in 2019, Pan et al. [23] solved the DPFSP with heuristic-based local search algorithms. In the same year, Ruiz et al. [24] proposed simplified iterated greedy heuristics to solve the DPFSP, while Meng et al. [25] developed the DPFSP to reduce the makespan under customer order constraints by the use of evolutionary and swarm-based optimization algorithms.

Consideration of environmental sustainability along with economic factors has recently appeared in the literature. In [26], Wang and Wang proposed, for the first time, an energy-efficient DPFSP to optimize the makespan and energy consumption simultaneously. Fu et al. [27] solved a stochastic energy-efficient DPFSP by a brain storm optimization heuristic. Wang et al. [8] developed a multi-objective whale optimization algorithm to solve the energy-efficient DPFSP. Last, but not least, Lu et al. [2] proposed the concept of a sustainable DPFSP, taking into consideration the energy consumption and a penalty coefficient for the process time. They defined this penalty function as a negative social factor. However, it does not meet the social sustainability criterion under the TBL and ISO 26000 [1]. They solved the problem using a multi-objective memetic optimization algorithm. It goes without saying that there are several other variants of the DPFSP [28-29], including for instance, the blocking DPFSP [4], preventive maintenance [9] and the no-wait DPFSP [10].

Table 1. Summary of the literature review for DPFSP studies

Paper	Year	Sustainability factors			Solution algorithm
		Economic	Environmental	Social	
Naderi and Ruiz [17]	2010	✓	-	-	Heuristics
Gao and Chen [18]	2011	✓	-	-	GALS
Lin et al. [19]	2013	✓	-	-	Modified iterated greedy search
Naderi and Ruiz [20]	2014	✓	-	-	Scatter search
Xu et al. [28]	2014	✓	-	-	Hybrid immune algorithm
Bargaoui et al. [21]	2017	✓	-	-	Chemical reaction algorithm
Fernandez-Viagas et al. [22]	2018	✓	-	-	Evolutionary search
Wang and Wang [26]	2018	✓	✓	-	Knowledge-based cooperative algorithm
Pan et al. [23]	2019	✓	-	-	Local search heuristic
Ruiz et al. [24]	2019	✓	-	-	Simplified iterated greedy search
Meng et al. [25]	2019	✓	-	-	Swarm-based evolutionary algorithm
Fu et al. [27]	2019	✓	✓	-	Brain storm optimization
Wang et al. [8]	2020	✓	✓	-	Whale swarm algorithm
Lu et al. [2]	2020	✓	✓	✓	Memetic optimization algorithm
Jing et al. [29]	2021	✓	-	-	Local search-based metaheuristics
Huang and Gu [30]	2021	✓	-	-	Biogeography-based optimization algorithm
This study	2021	✓	✓	✓	Learning-based SEO

					with local search

Table 1 presents a summary of the literature review and collects all the DPFSPs related to sustainability factors, including economic, environmental and social factors, as well as the solution algorithm. From this table, the following research gaps can be identified:

- Only one study [2] has considered the triple bottom line concept in modeling a sustainable DPFSP. However, the job opportunity and lost working days were not considered.
- No study has applied an SEO or any version of this algorithm to the area of DPFSP.

In terms of research gaps, only one study [2] considered social factors using a penalty coefficient associated with the task process times. However, as can be seen in the ISO 2600 guidelines respecting social responsibility in the production and supply chain systems to improve humans' life quality [35], job opportunities and lost working days are two of the main factors that must be considered in order to achieve social sustainability. In this regard, a novel multi-objective MILP is developed to minimize the makespan and energy consumption while maximizing social benefits. To get our DPFSP closer to real production systems such as those in the Canadian wood industry, the proposed problem assumes that centers are non-identical as they handle different forest products. This problem also considers different technologies for machines which have a high impact on environmental and social factors. New technologies can increase the speed of operation of tasks, but at the cost of increasing the energy consumed and creating fewer job opportunities for workers in comparison to traditional production systems. To the best of the authors' knowledge, no similar study has considered these items simultaneously in order to establish a sustainable DPFSP. Another novelty of this paper is the development of a new optimizer for solving our mathematical model. This study proposes a new SEO version [16] created with the help of learning-based operators and local search-based heuristics to solve our multi-objective optimization problem. The proposed algorithm is able to generate higher-efficient Pareto-based solutions than what are obtained with the general version of SEO and other state-of-the art methods in the literature.

3. Proposed problem

This section starts by defining the notations used for the mathematical modeling of the proposed sustainable DPFSP as follows:

Indices:

f	Index of production centers, $f = \{1, 2, \dots, F\}$
m	Index of machines in each center, $m = \{1, 2, \dots, M\}$
n	Index of tasks, $n = \{1,2, \dots, N\}$
t	Index of production technologies, $t = \{1,2, \dots, T\}$
i	Index of task positions in a schedule, $i = \{1,2, \dots, N\}$

Parameters:

B	Maximum budget for the establishment of machines and technologies with the salary of workers for all the production centers
CO_{mtf}	Cost for the implementation of machine *m* with technology *t* in the production center *f*
JO_{mtf}	Job opportunities created by the use of machine *m* with technology *t* in the production center *f*

CJ_{mtf}	Salary of operators working on machine *m* with technology *t* per unit of time in the production center *f*
LD_{mtf}	Lost days due to the use of machine *m* with technology *t* in the production center *f*
MW	Maximum allowable ratio of broken products in all the production centers
RW_{mtf}	Ratio of broken products when machine *m* and technology *t* are used in the production center *f*
O_{nmtf}	Operation of task *n* on machine *m* with technology *t* in the production center *f*
P_{nmtf}	Process time of operation O_{nmtf}
IEC_{mtf}	Idle energy consumption of machine *m* with technology *t* per unit of time in the production center *f*
UEC_{mtf}	Useful energy consumption of machine *m* with technology *t* per unit of time in the production center *f*
EC_{mtf}	Energy consumption due to implementing machine *m* with technology *t* per unit of time in the production center *f*
WJ	Weight of job opportunities
WL	Weight of lost working days

Decision variables:

A_f	Number of tasks assigned to the production center *f*
Y_{mtf}	If the machine *m* is using the technology *t* in the production center *f*, 1; otherwise, 0
ST_{imtf}	Starting time of the task at position *i* on machine *m* using technology *t* in the production center *f*
X_{nimtf}	If the task *n* is set at position *i* on machine *m* with the use of technology *t* in the production center *f*, 1; otherwise, 0
T_{nmtf}	Idle time of the operation of task *n* on machine *m* with technology *t* in the production center *f* (O_{nmtf})
C_{imtf}	Completion time of a task at position *i* on machine *m* with technology *t* in the production center *f*
CT_f	Time for completing tasks in the production center *f*
C_{max}	Maximal completion time for all the production centers

From the description of the proposed problem, there are N tasks distributed across F non-identical production centers. Each center has M different machines with T technologies and follows a PFSP conceptually. For each task, there are O operations. These operations are handled one by one for the assigned production center. All the production centers are able to perform all the tasks. When the scheduling starts, all the machines and centers are available. After a task is assigned to a production center, the task must be processed at that center, and cannot be transferred to another one. No interruption is allowed in the proposed production system. The process time (P_{nmtf}) for the operation (O_{nmtf}) of the tasks is different based on the production centers, machines and technologies. For each machine, there are some technologies which change the speed of the operations, the energy consumption and the social factors linked to job opportunities for operators, and work days lost while learning this technology and updating the workers' knowledge. Next, the criteria of TBL used in the definition of our problem, i.e., economic sustainability, environmental sustainability and social sustainability, are illustrated followed by our proposed mathematical model.

3.1. Economic sustainability

In most production scheduling models, the makespan (C_{max}) is the only economic criterion [13-14, 24-25, 28-30]. This criterion reflects the benefit of a production system or its economic value. The present study is not limited to the achievement of economic sustainability. It considers not only the makespan, but also the worker salaries and the production rates of technologies used. Let us assume that a company supports the total cost of a production system and has a maximum budget (B). This company must consider the costs of purchasing machines with new technologies (CO_{mtf}). Workers' salaries also vary with the production centers, the machines and the technologies (CJ_{mtf}). Only one production technology must be implemented on each machine. Wastes are different for each technology (RW_{mtf}), and with regards to the concept of economic sustainability, the maximum allowable wastes must be met by these machines and technologies (MW).

3.2. Environmental sustainability

Environmental pollution in production operations is certainly the main culprit in the context of global warming and climate change in developed countries like Canada. To control environmental pollution in production systems and supply chains [31-33], the International Organization for Standardization (ISO) proposed the ISO 14000 standard for environmental sustainability management [34]. Having cleaner production is particularly a concern in the lumber industry, considered to be the leader of environmental pollution in Canada. Compared to traditional technologies which are generally based on the use of non-renewable energies, new production technologies consume fewer non-renewable energy resources. In this regard, recent studies proposed the energy-efficient DPFSP as a solution for this challenge [26-27]. To achieve environmental sustainability, this study not only considers the energy consumption of working time (UEC_{mtf}) and idle time (IEC_{mtf}), but also the energy consumption related to the implementation of the technology of turning the machines on and off (EC_{mtf}).

3.3. Social sustainability

Social sustainability involves many factors linked to the work environment, healthcare and social development. In the ISO 26000 standard used by governments and business networks to achieve social responsibility [35], there is a guideline, the SA8000, that considers the job opportunities and lost days for injuries of workers [36]. It should be noted that social sustainability is not limited to only these two factors as consumer risk and local business development are two common criteria used in relevant works [37-38].

For the first time in the area of DPFSP, the number of operators working on a machine using a particular technology in a production center is considered explicitly (JO_{mtf}). The number of work days lost due to the implementation of a new production technology on a machine (LD_{mtf}) is considered as another social factor. The lost working days represent in fact the time needed to teach operators this new technology. These social factors are weighted (WJ and WL) in the third objective function which aims to achieve social sustainability.

3.4. Mathematical model

Generally, the proposed sustainable DPFSP aims to find, for each production center, the optimal number of tasks allocated (A_f), the time needed to complete the tasks (CT_f), the optimal allocation of technologies to machines (Y_{mtf}), the optimal sequence of tasks (X_{nimtf}) and other optimal values of the decision variables defined earlier. The proposed mathematical model is described as follows:

$$Z_1 = \min_{Economic}(C_{max}) \tag{1}$$

$$Z_2 = \min_{Environmental}\Big(\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}(Y_{mtf} \times EC_{mtf}) + \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}\sum_{n=1}^{N}\sum_{i=1}^{N}(X_{nimtf} \times UEC_{mtf} \times P_{nmtf}) + \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}\sum_{n=1}^{N}(T_{nmtf} \times IEC_{mtf})\Big) \tag{2}$$

$$Z_3 = \max_{Social}\Big(WJ \times \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}(Y_{mtf} \times JO_{mtf}) - WL \times \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}(Y_{mtf} \times LD_{mtf})\Big) \tag{3}$$

$s.t.$

$$\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}(Y_{mtf} \times JO_{mtf} \times CJ_{mtf}) + \sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}(Y_{mtf} \times CO_{mtf}) \le B \tag{4}$$

$$\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}(Y_{mtf} \times RW_{mtf}) \le MW \tag{5}$$

$$\sum_{i=1}^{N}\sum_{f=1}^{F}X_{nimtf} = 1, \qquad \forall n \in N, m \in M, t \in T \tag{6}$$

$$\sum_{n=1}^{N}\sum_{f=1}^{F}X_{nimtf} = 1, \qquad \forall i \in N, m \in M, t \in T \tag{7}$$

$$\sum_{n=1}^{N}\sum_{i=1}^{N}\sum_{m=1}^{M}\sum_{t=1}^{T}(X_{nimtf}) = A_f, \qquad \forall f \in F \tag{8}$$

$$\sum_{n=1}^{N}\sum_{i=1}^{N}X_{nimtf} = Y_{mtf}, \qquad \forall m \in M, t \in T, f \in F \tag{9}$$

$$\sum_{t=1}^{T}Y_{mtf} = 1, \qquad \forall m \in M, f \in F \tag{10}$$

$$\sum_{i=1}^{N}ST_{imtf} \le \left(\sum_{i=1}^{N}\sum_{m=1}^{M}\sum_{t=1}^{T}\sum_{f=1}^{F}P_{nmtf}\right) \times Y_{mtf}, \forall m \in M, t \in T, f \in F \tag{11}$$

$$C_{imtf} = ST_{imtf} + \sum_{n=1}^{N}(X_{nimtf} \times P_{nmtf}), \forall i \in N, m \in M, t \in T, f \in F \tag{12}$$

$$C_{imtf} \ge ST_{i,m-1,tf} + \sum_{n=1}^{N}(X_{nimtf} \times P_{nmtf}), \forall i \in N, m > 1, t \in T, f \in F \tag{13}$$

$$C_{imtf} \ge ST_{i-1,mtf} + \sum_{n=1}^{N}(X_{nimtf} \times P_{nmtf}), \forall i > 1, m \in M, t \in T, f \in F \tag{14}$$

$$T_{nmtf} = \sum_{i>1}^{N}(C_{imtf} - C_{i-1,mtf} - (X_{nimtf} \times P_{nmtf})), \forall n \in N, m \in M, t \in T, f \in F \tag{15}$$

$$CT_f \geq \sum_{i=1}^{I} \sum_{m=1}^{M} \sum_{t=1}^{T} C_{imtf}, \qquad \forall f \in F \tag{16}$$

$$C_{max} \geq CT_f, \qquad \forall f \in F \tag{17}$$

$$A_f, ST_{imtf}, C_{imtf}, CT_f, C_{max}, T_{nmtf} \geq 0 \tag{18}$$

$$Y_{mtf}, X_{nimtf} \in \{1,0\} \tag{19}$$

Equations (1) to (3) represent the objective functions which are limited by constraints (4) to (19). The optimal solution is found by minimizing the makespan (Equation (1)) and the energy consumption (Equation (2)) while maximizing the social benefits (Equation (3)). The energy consumption includes the energy required to implement a technology in a machine, the energy used to process a task on a machine and the energy consumed by a machine during an idle period of time when a task is pending. The social benefits include two terms, job opportunities and lost working days.

The constraint set (4) concerns the maximum budget available to cover the salary of the operators and the cost associated with the implementation of technologies on the machines. The maximum ratio of broken products or waste authorized in all production centers is considered in the set of constraints (5). Constraints (6) and (7) show that each task must have a unique schedule. The constraint set (8) guarantees that the required number of tasks is assigned in each production center. The constraint set (9) shows the relationship between the allocation of tasks to production centers and the technology selection for machines. The constraint set (10) ensures that each machine is assigned to one technology. The constraint set (11) relates the start time of tasks to the total operating time of all machines in all production centers. The constraint set (12) shows that the full time of a task is defined by its start and processing times. Constraints (13) and (14) show the relationship between machine schedules and tasks in a sequence. Constraint (15) computes the idle time of the machines. The constraint set (16) limits the maximum time allowed to complete all tasks in a production center while the constraint set (17) ensures that the makespan of all production centers is less than or equal to the maximum allowable execution time. Finally, the constraints (18) and (19) define the feasible set of values of the decision variables in the model.

3.5. Numerical example

In order to show that the proposed optimization model has a feasible solution and to numerically illustrate the proposed sustainable DPFSP, an example with 4 tasks (J_1, J_2, J_3 and J_4), 2 production centers (F_1 and F_2), 2 machines (M_1 and M_2) and 2 technologies (T_1 and T_2), is provided. Table 2 provides the data used for task processing time, cost of machines, job opportunities, lost days, operators salaries, idle and utile energy consumption rate, and energy consumption for switching machines on and off. The maximum budget of the company is set to 0.5 million dollar and the maximum allowable number of broken products is set to 30 percent in this example. Finally, the weights for social factors, including job opportunities and lost working days, are set at 0.9 and 0.1 respectively.

Table 2. Processing time of tasks and other parameters values

Tasks	Unit	Production center F_1	Production center F_2

		Machine M_1		Machine M_2		Machine M_1		Machine M_2	
		T_1	T_2	T_1	T_2	T_1	T_2	T_1	T_2
J_1	Hour	4	5	6	5	3	4	6	6
J_2	Hour	3	6	4	4	4	6	3	4
J_3	Hour	5	4	2	2	6	5	2	3
J_4	Hour	2	3	4	6	3	4	5	4
Implementation cost (CO_{mtf})	$	12×10^4	15×10^4	13×10^4	9×10^4	12×10^4	14×10^4	11×10^4	12×10^4
Job opportunities (JO_{mtf})	Person	2	3	4	2	4	6	3	5
Salary of operators (CJ_{mtf})	$	10	8	12	10	10	9	12	8
Lost days (LD_{mtf})	Days	14	10	21	14	10	14	12	8
Ratio of broken products (RW_{mtf})	Scalar	0.08	0.02	0.05	0.09	0.05	0.07	0.06	0.07
Idle energy consumption rate (IEC_{mtf})	BTU per hour (*[3])	8.98×10^5	9.8×10^5	10.2×10^5	8.75×10^5	8.75×10^5	8.55×10^5	8.25×10^5	8.6×10^5
Utile energy consumption rate (UEC_{mtf})	BTU per hour	5.36×10^5	5.4×10^5	6.4×10^5	5.2×10^5	4.6×10^5	3.6×10^5	5.1×10^5	5.5×10^5
Energy consumption for implementation (EC_{mtf})	BTU	30.2×10^5	28.4×10^5	26.2×10^5	25.4×10^5	27.2×10^5	26.8×10^5	24.8×10^5	26.2×10^5

This numerical example has an optimal solution confirming the feasibility of the proposed optimization model. This solution allocates the first technology (T_1) to the first machine (M_1) in the first production center (F_1). In the same center, the second technology (T_2) is assigned to the second machine (M_2). In the second production center (F_2), the first technology is selected to be used for both machines. Therefore, the decision variables are $Y_{111} = Y_{221} = Y_{112} = Y_{212} = 1$ while the others are zero.

[3] British Thermal Unit (BTU), note that 10^{15} BTU equals to 1055×10^{18} joules

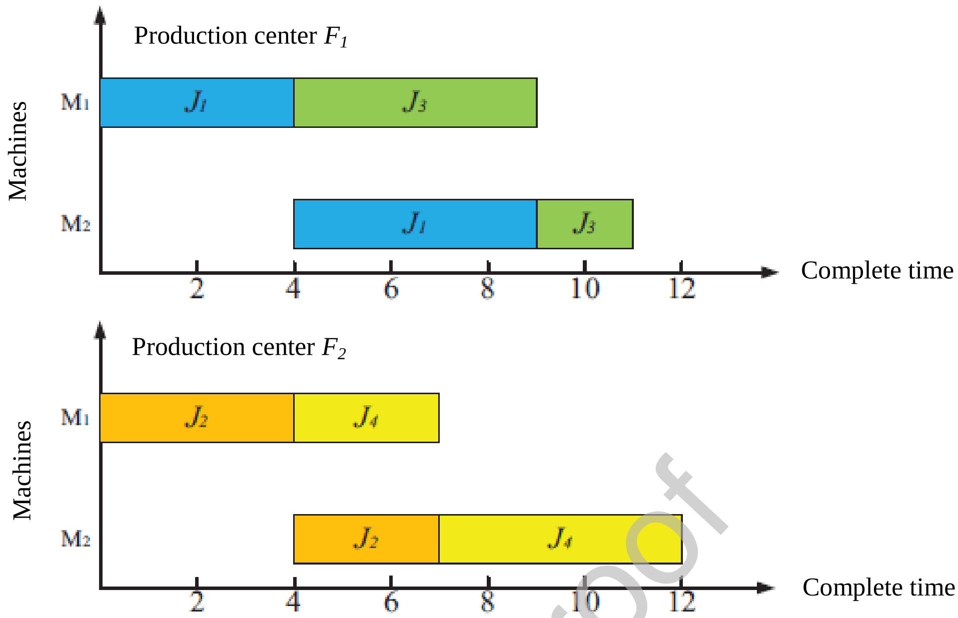


Figure 1. Presentation of the optimal solution

The permutation of the tasks [J_1, J_2, J_3, J_4] of the optimal solution is shown in Figure 1. The tasks J_1 and J_3 are assigned to the first production center, while the tasks J_2 and J_4 are assigned to the second production center. The completion times are 11 and 12 hours respectively in the first and second centers. Based on these outputs, the makespan is 12 hours, the total energy consumption is 33 324 000 BTUs and finally the social criteria value is 4.9.

4. Proposed algorithm

As mentioned earlier, the classical version of the DPFSP is NP-hard. The proposed DPFSP which includes three conflicting objectives and real-life constraints such as, for example, the maximal budget, is more complex than most of the existing studies. To tackle this optimization model, this study develops a new heuristic that is an extension of the recently proposed SEO approach [16], includes learning-based operators and local search-based techniques and is able to solve multi-objective problems.

The SEO algorithm was chosen as the base method because of its high computational time efficiency in solving NP-hard problems such as routing optimization [33] and truck scheduling problems [39]. Furthermore, although many recently developed optimizers such as the immune [28], chemical reaction [21], whale swarm [8], and brain storm [27] algorithms have been studied in the field of DPFSP, to the best of our knowledge, no study has applied SEO so far in this area.

The flowchart shown in Figure 2 presents the general framework of the original SEO algorithm.

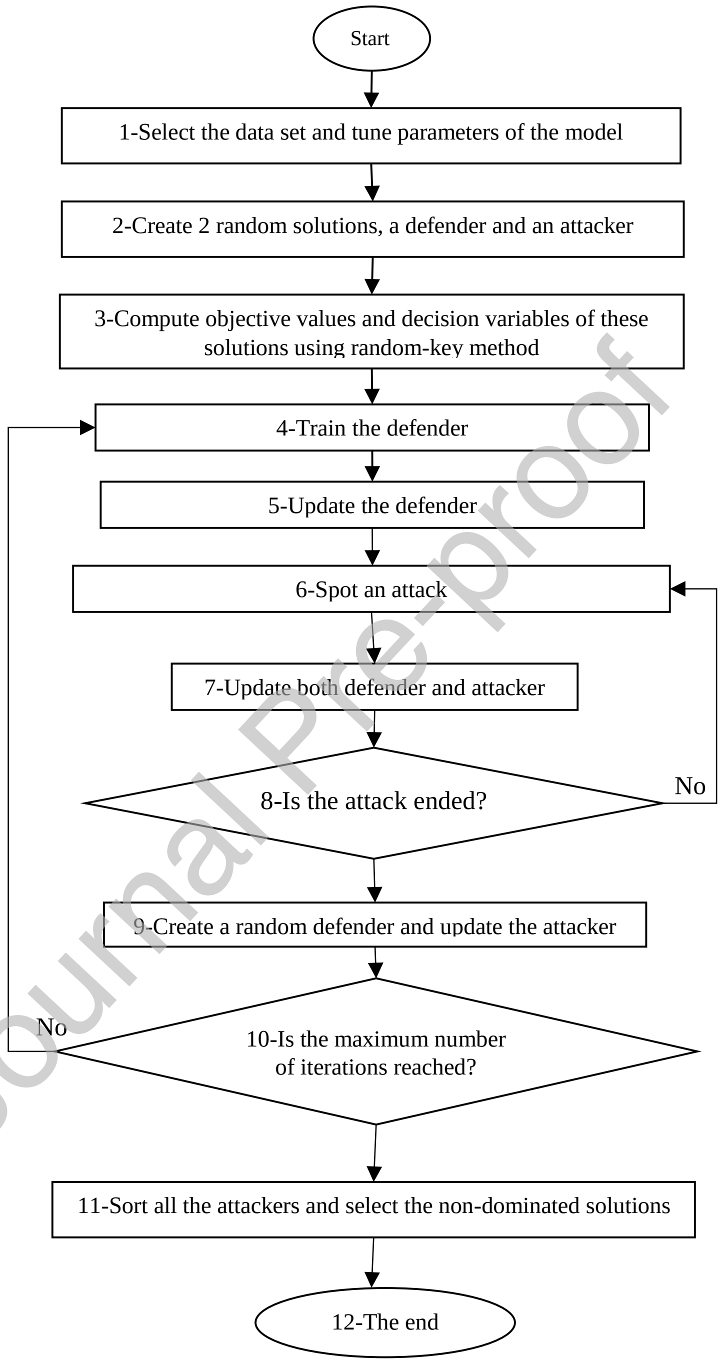


Figure 2. Flowchart of the original SEO algorithm

Based on social engineering rules, the approach starts with two initial solutions determined randomly (step 2). The dominant solution, or, in other words, the solution with the best values in at least two objectives, is selected as the attacker while the other solution is set as the defender. To train the defender (step 4), the attacker copies a percentage of the defender. If this newly trained defender is dominant, the last defender is updated with this new one (step 5). Then, the social engineering attacks start (step 6): new attackers and defenders are generated and compared to the previous ones. If a new defender is able to dominate the attacker, their positions are swapped (step7). Finally, the best solution is saved as the attacker and sent to the list of Pareto solutions while a new random solution replaces the defender (step 9). Once the maximum number of iterations has been reached (step 10), the list of Pareto solutions is sorted and the non-dominated solutions are found (step 11).

The main difference of the proposed algorithm with the original multi-objective SEO is in steps 4 and 6 where a learning-based operator and a local search technique are developed to improve the performance of SEO.

4.1.Encoding and decoding schemes

Encoding and decoding schemes are vital to heuristically solve a mathematical model comprising different optimization objectives and constraints [40-41]. While the encoding is performed by the SEO-based algorithm (step 2), the random key method [42] is used as the decoding scheme applied in steps 3, 5 and 7.

In the proposed DPFSP, the decoding scheme is used for three purposes: (*i*) the selection of production technology for each machine; (*ii*) the allocation of tasks for each production center and, (*iii*) the scheduling of tasks on the machines of each center. The pseudo-code provided in Figure 3 shows how the technology is selected for a machine under budget and waste ratio constraints. The matrix received from the main algorithm contains input variables whose values vary between zero and one. The random key method selects the minimum array within this matrix that meets the constraints on budget, waste and technology selection (respectively constraints (4), (5) and (10)). If there is no minimum array meeting the 3 constraints, the technology with the lowest cost (constraint 4) or the lowest waste ratio (constraint 5), is selected.

```
X; %Input received from the main algorithm
model(); %Input data function
M=model.M; %Number of machines
F=model.F;    %Number of production centers
T=model.T;   %Number of technologies
JOmtf=model.JOmtf;   %Job opportunities
CJmtf=model.CJmtf;    %Salary of operators
RWmtf=model.RWmtf;  %Ratio of broken products
COmtf=model.COmtf;    %Cost of implementation
B=model.B;       %Budget
MW=model.MW;  %Maximum broken products  ratio
%% Loop for the technology selection
Ymtf=zeros(M,T,F);    %Decision variables? for technology selection
BB=0;     %Counter
MWW=0;    %Counter
XX=X;   %Selection from all the variables
m=1;           %Counter
f=1;           %Counter
while BB<=B && MWW<=MW && m<=M && f<=F
            [a, b]=min(XX(:,m));
            Ymtf(m,b,f)=1;
            BB=BB+Ymtf(m,b,f)*CJmtf(m,b,f)*JOmtf(m,b,f)+ COmtf(m,b,f)*Ymtf(m,b,f);
```

```
            MWW=MWW+Ymtf(m,b,f)*RWmtf(m,b,f);
            m=m+1;
            f=f+1;
end
for  m=1:M
     for f=1:F
        if   BB<=B && sum(Ymtf(m,:,f))==0
            [a, b]=min(COmtf(m,:,f));
            Ymtf(m,b,f)=1;
             BB=BB+Ymtf(m,b,f)*CJmtf(m,b,f)*JOmtf(m,b,f)+ COmtf(m,b,f)*Ymtf(m,b,f);
        end
        if   MWW<=MW && sum(Ymtf(m,:,f))==0
            [a, b]=min(COmtf(m,:,f));
            Ymtf(m,b,f)=1;
             MWW=MWW+Ymtf(m,b,f)*RWmtf(m,b,f);
        end
     end
end
Ymtf;
```

Figure 3. Pseudo-code describing the allocation of a technology to each machine

Arrays generated by the random-key algorithm that contain values between zero and one are sorted to provide the sequence of tasks. The assignment of tasks to production centers and machines is then carried out according to the feasibility of the sequence of tasks with regard to the selected technology. Figure 4 shows the pseudo-code of these last two decoding phases.

```
Y;        %Input taken from the main algorithm
model(); %Input data function
M=model.M; %Number of machines
F=model.F;    %Number of production centers
T=model.T;   %Number of technologies
Ymtf;    %Decision variable for technology selection
N=model.n;    %Number of tasks
Pnmtf=model.Pnmtf;   %Process time
%% Loop for the task scheduling
Xnimtf=zeros(N, N, M,T,F);   %Decision variables for scheduling
Cimtf=zeros(N,M,T,F);           %Complete time for each position
[a, b]=sort(Y);
for  m=1:M
     for f=1:F
        for t=1:T
            if Ymtf(m,t,f)==1
               for n=1:N
                  bb=b(n)
                   Xnimtf(bb,n,m,t,f)=1;
                   Cimtf(n,m,t,f)=Cimtf(n,m,t,f)+ Xnimtf(bb,n,m,t,f)*Pnmtf(bb,m,t,f);
               end
            end
        end
     end
end
Xnimtf;
```

Figure 4. Pseudo-code describing the scheduling of tasks

4.2. Learning-based SEO

Recently, many modifications and hybridization of heuristic algorithms using learning-based concepts and local search-based operators have been proposed [45-48]. Like other random-based heuristics, SEO includes exploration and exploitation phases [41-44]. The exploitation phase involves training and retraining of the defenders [44] while social engineering attacks force the exploration of new solutions [33]. The main contribution of this study is the creation of new operators to improve the quality of the non-dominated solutions and to reduce the computation time of the algorithm.

This results in the creation of the Learning-based SEO (LSEO) which dynamically updates the training ratio of the defender (*Alpha*). This automatic updater makes it easier for users to implement the algorithm. After trying a few values for each parameter, the algorithm updates the parameter values as follows:

$$Alpha_{It} = \begin{cases} Lower_{Alpha} + (Upper_{Alpha} - Lower_{Alpha})\left(\frac{It}{MaxIt}\right), \text{ If the new defender dominates the current} \\ Alpha_{It} = Upper_{Alpha} - (Upper_{Alpha} - Lower_{Alpha})\left(\frac{It}{MaxIt}\right), \text{ Otherwise} \end{cases} \quad (20)$$

where $Alpha_{It}$ is the value of *Alpha* at iteration *It*, $Upper_{Alpha}$ and $Lower_{Alpha}$ are upper and lower bounds of this parameter, respectively, and *MaxIt* is the maximum number of iterations.

This study also proposes a new methodology for carrying out a local search. With each attack, a new defender and attacker are generated using the following formulas:

$$Defender_{new} = \frac{Defender_{old} + Attacker_{old}}{2} + rand \times \left((Upper_{bound} - Lower_{bound}) * rand\right) + Lower_{bound}) \quad (21)$$

$$Attacker_{new} = \frac{Defender_{old} + Attacker_{old}}{2} - rand \times \left((Upper_{bound} - Lower_{bound}) * rand\right) + Lower_{bound}) \quad (22)$$

where $Defender_{old}$ and $Attacker_{old}$ are respectively the defender and attacker before the attack, while $Defender_{new}$ and $Attacker_{new}$ are respectively the defender and attacker after the attack, and *rand* is a value chosen randomly between zero and one. Furthermore, $Upper_{bound}$ and $Lower_{bound}$ are respectively the upper and lower bounds of this new way of exploring the search area. The proposed multi-objective LSEO algorithm is summarized in Figure 5.

```
MaxIt;        %Maximum number of iteration
Nat;          %Number of attacks
Upper_Alpha;     %Maximum training ratio
Lower_Alpha;     %Minimum training ratio
%% Main loop
Create two solutions;
Sort the solutions and select the better one as the attacker;
Another solution is selected as the defender.
t=1;        % Counter
List;       %List of Pareto solutions
while  t≤ MaxIt
    Do the training using equation (20).
    nt=1;
    while nt ≤ Nat
        nt=nt+1;
        Select the technique given in equations (21) and (22) to do an attack;
        Update the defender and the attacker if they can dominate the previous one;
     end
     Exchange the defender and attacker if the defender is able to dominate the attacker;
     Send the attacker to the List;
     Create a new random solution as the defender;
     t=t+1;
end
 Evaluate the List and generate the Pareto fronts;
```

Select the non-dominated solutions and show them;

Figure 5. Pseudo-code of the multi-objective LSEO

5. Computational results

In this section, the proposed industrial example and the simulated test studies used to do our analyses are first provided. Then, different criteria and metrics used to assess the algorithms are defined and the parameter values are tuned leading to a fair comparison of the provided algorithms. Next, a validation study is performed to find the exact solution using an epsilon-constraint method. Then, the performance of our algorithms is compared to that of various traditional and recent algorithms using evaluation metrics. The robustness of the proposed optimization model is also evaluated by a sensitivity analysis. It should be noted that our codes were written in GAMS and MATLAB software and implemented on a laptop with 1.7 GB CPU and 6.0 GB RAM.

5.1.Industrial example and tests

Canadian Wood Products (CWP)[4] is one of the well-known practitioners of the wood industry in Canada. This company is the leader in the production and distribution wood products in North America. It has three main products, including softwoods, industrial and architectural lumbers. For each, a specific production technology must be installed on the machines. Moreover, six operational tasks are required including cutting, custom processing, drying, classifying, storing and loading. Last but not least, the CWP has three main production and distribution centers in Buffalo, Montreal and Concord.

The industrial example of the company CWP is used to show the applicability of the optimization method developed. For security reasons, the actual values of the parameters are not accessible and therefore, estimated values are provided. Moreover, in order to evaluate our approach with 3 levels of model complexity, 12 tests were created, 4 tests for each level of complexity, namely, small, medium and large models as shown in Table 3.

Table 3. Test studies used to evaluate the proposed algorithm

Complexity level of the model	Number of test studies	Number of centers (*F*)	Number of machines (*M*)	Number of technologies (*T*)	Number of tasks (*N*)
Small Size	Industrial example	3	3	3	6
	T1	2	2	2	4
	T2	2	2	2	8
	T3	2	4	2	20
	T4	3	4	3	30
Medium Size	T5	3	6	2	30
	T6	3	6	3	40
	T7	4	8	4	30
	T8	4	8	5	40

[4] https://canadianwood.ca/

Large Size	T9	6	12	4	80
	T10	6	12	5	100
	T11	8	16	6	80
	T12	10	16	6	100

Since the optimization model proposed for a sustainable DPFSP is novel depending on the different production centers and technologies as well as social factors, there is no benchmark dataset available corresponding to our optimization model. In this regard, the possible value ranges for the model parameters are presented in Table 4. To fix the parameters values, we run random functions for each test size and then save the values.

Table 4. Ranges of values for model's parameters

Parameter	**Range**
P_{nmtf}	*randi*([2, 8],N, M, T, F)
CO_{mtf}	*randi*([8, 20],M, T, F)*10^4
JO_{mtf}	*randi*([2, 9],M, T, F)
CJ_{mtf}	*randi*([8, 20],M, T, F)
LD_{mtf}	*randi*([8, 30],M, T, F)
WJ	0.9
WL	0.1
RW_{mtf}	*rand*(M, T, F)*0.1
IEC_{mtf}	(*randi*([8, 12],M, T, F)+*rand*())*10^5
UEC_{mtf}	(*randi*([2, 7],M, T, F)+*rand*())*10^5
EC_{mtf}	(*randi*([20, 40],M, T, F)+*rand*())*10^5
B	*randi*([*round*(*sum*(JO_{mtf}.* CJ_{mtf} + CO_{mtf})/2), *round*(*sum*(JO_{mtf}.* CJ_{mtf} + CO_{mtf}))
MW	**if** *sum*(RW_{mtf})>1 *randi*([*round*(*sum*(RW_{mtf})/2), *round*(*sum*(RW_{mtf})) **else** *rand*()+(*sum*(RW_{mtf})/2) **end**

**randi* is a function which generates random integer numbers between lower and upper bounds.
**rand* is a function which generates random continuous numbers between zero and one.
**round* is a function which transforms a continuous number to the closest integer number.
**sum* is a function to sum numbers contained in a matrix.

5.2. Assessment metrics and parameter tuning

As mentioned earlier, this study develops LSEO as an improvement to SEO. This study not only compares the performance of LSEO with that of SEO but also with the performance of other well-known and state-of-the-art algorithms in the literature. In this regard, the Non-dominated Sorting Genetic Algorithm (NSGA-II) [49], the enhanced Strength of Pareto Evolutionary Algorithm (SPEA2) [50], the Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) [51] as well as two recent algorithms comprising the Multi-Objective

Brain Storm Optimization (MOBSO) [27] and the Multi-Objective Keshtel Algorithm (MOKA) [52] are used.

These algorithms are evaluated using assessment metrics. In addition to the algorithm computation time criterion (CPU time), the Number of Pareto Solutions (NPS) [53], the Mean Ideal Distance (MID) [54], the Maximum Spread (MS) [55] and the Hypervolume (HV) [56] are considered to evaluate the Pareto solutions found by the algorithms. These metrics are defined hereafter:

- NPS is the number of non-dominated solutions in the Pareto optimal set. A higher value of this metric shows a better diversity of the solutions [53].
- MS measures the distance between the best and the worst solutions in the optimal Pareto set. It can be formulated as follows:

$$MS = \sqrt{\left(\sum_{j=1}^{NO}(Z_j^{Max} - Z_j^{Min})\right)^2} \tag{23}$$

where Z_j^{Max} and Z_j^{Min} are respectively the maximum and the minimum value of the objective j among all the solutions Along with the NPS metric, this metric evaluates the diversity of the solutions. A higher value of the MS metric means a better capability of the algorithm [55] to find an optimal solution.

- MID measures the distance between solutions in the Pareto optimal set and we can formulate it as follows:

$$MID = \frac{\sum_{i=1}^{NPS}\sqrt{\left(\sum_{j=1}^{NO}\frac{Z_j^i - Z_j^{Best}}{Z_j^{Max} - Z_j^{Min}}\right)^2}}{NPS} \tag{24}$$

where NO is the number of objectives, Z_j^i is the solution i for objective j, and Z_j^{Best} is the maximum or minimum value with regards to the type of the objective function. A lower value of this metric shows a faster convergence of the solution [54].

- HV computes the space of non-dominated solutions. It is difficult to calculate HV exactly as it cannot be formulated mathematically. An approximation method such as Monte Carlo is usually used to compute this metric. In this study, the simulation method of Zitzler et al., [56] was used to quantify HV. A higher value of this metric shows a better performance of the Pareto set.

With regard to the above-mentioned criteria, the algorithms must be tuned before the validation and comparison studies [57-58]. Good tuning helps algorithms achieve their best performance and therefore, leads to a fair comparison [59-60]. Consequently, in this study, the parameters were tuned using the Taguchi method [61]. Taguchi first uses orthogonal arrays to reduce the number of experiments using only selected experiments. For example, if an algorithm has five parameters and each has three candidate values, the total number of experiments for one run is 3^5 =243. However, Taguchi uses an orthogonal array of L_{27} reducing the number of experiments to 27. Taguchi is also based on noise and control factors which are evaluated by the Signal to Noise (S/N) ratio and Relative Percentage Deviation (RPD), respectively. In the context of multi-objective optimization, assessment metrics are used. The S/N ratio can be formulated as:

$$S/N = -10 \times \log_{10}\left(\frac{\sum_i HV_i^2}{n}\right) \tag{25}$$

where n is the number of orthogonal arrays and HV_i is the response value of the i^{th} orthogonal array. Similar to the *HV* metric, a higher value of S/N is preferable for this noise factor. As shown in the following formula, the control factor evaluated by RPD includes *MID* and *MS* metrics that quantify the precision and diversity of Pareto solutions, respectively:

$$RPD = \frac{MID}{MS} \tag{26}$$

Thus, a lower value of *RPD* means better performance of the algorithm.

In order to make an unbiased comparison, the maximum number of fitness evaluations is set to the same values for all algorithms under evaluation based on the size of the model's complexity levels: 25000, 50000 and 100000 for small, medium and large sizes, respectively. Therefore, for SEO and LSEO as one-solution algorithms, the maximum number of iterations (*MaxIt*) and maximum number of attacks (*Nat*) are set to 500 and 50 respectively for small sizes (500×50 = 25000), to 1000 and 50 for medium sizes (1000×50 = 50000) and up to 2000 and 50 for large sizes (2000×50 = 100000). For population-based algorithms including NSGA-II, SPEA2, MOEA/D, MOBSO and MOKA, the maximum number of generations (*MaxIt*) and population size (*nPop*) are respectively set to 250 and 100 for small sizes (250×100 = 25000), to 500 and 100 for medium sizes (500×100 = 50000) and to 1000 and 100 for large sizes (1000×100 = 100000). Other parameters of each algorithm were adjusted based on candidate values identified in previous studies [27, 39-42, 51-52] as reported in Table 5.

Table 5. Candidate values for parameters of algorithms under evaluation

Algorithms	Parameters	Candidate values		
SEO	Percentage of training (*Alpha*)	0.1	0.3	0.5
	Rate of attack (*Betta*)	0.05	0.15	0.25
LSEO	Upper bound of *Alpha* ($Upper_{Alpha}$)	0.8	0.9	1
	Lower bound of *Alpha* ($Lower_{Alpha}$)	0	0.1	0.2
NSGA-II	Percentage of crossover (P_C)	0.5	0.6	0.7
	Percentage of mutation (P_m)	0.1	0.15	0.2
SPEA2	Number of archive (N_A)	50	75	100
MOEA/D	Number of subproblems considered in MOEA/D (*N*)	150	200	250
	Number of weight factors (*T*)	12	25	50
MOBSO	Probability of each generation (P_g)	0.2	0.4	0.8
	Probability of first cluster (P_{c1})	0.2	0.4	0.6
	Probability of second cluster (P_{c2})	0.2	0.4	0.6
MOKA	Number of swirling (*NS*)	2	3	5
	Percentage of lucky Keshtels (*N1*)	0.1	0.2	0.4
	Percentage of moving Keshtels (*N2*)	0.4	0.5	0.6
	Percentage of random Keshtels (*N3*)	$N3 = 1 - N1 - N2;$		

The orthogonal array for SEO, LSEO, NSGA-II and MOEA/D is a full factorial method (3×3 = 9). As such, SPEA2 has three tests. The orthogonal array of L_9 is used for MOBSO and MOKA. Based on the calculation of the noise and the control factors, the best candidate value for each parameter is reported in Table 6.

Table 6. Tuned parameters of the algorithms

Algorithm	Parameters
SEO	*Alpha*=0.3; *Beta*=0.05;
LSEO	$Upper_{Alpha} = 1$; $Lower_{Alpha} = 0.1$;
NSGA-II	$P_C = 0.7$; $P_m = 0.1$;
SPEA2	$N_A = 100$;
MOEA/D	$N = 200; T = 50$;
MOBSO	$P_g = 0.8; P_{c1} = 0.4$; $P_{c2} = 0.2$;
MOKA	$NS = 3; N1 = 0.2; N2 = 0.5$; $N3 = 0.3$;

5.3.Validation

The Epsilon-Constraint (EC) method [62] is solely used to find exact solutions to our example of an industrial problem in order to validate the performance of the proposed algorithms. This algorithm optimizes one main objective and uses upper and lower bounds for other objective functions. As the economic criterion is generally more important than environmental and social criteria for production managers, the first objective is chosen in this study as the main objective. Therefore, the problem addressed by the EC method can be formulated as follows:

$$\begin{aligned} &\min_{Economic} Z_1 \\ &\textit{s.t.}\ \text{Constraints (4) to (19)} \\ &Z_2 \leq EC_2 \\ &Z_3 \geq EC_3 \\ &Z_2^{Min} \leq EC_2 \leq Z_2^{Max} \\ &Z_3^{Min} \leq EC_3 \leq Z_3^{Max} \end{aligned} \tag{27}$$

where EC_2 and EC_3 are allowable bounds of the second and third objectives, respectively, while the lower and upper bounds for the second objective are $Z_2^{Min}$and Z_2^{Max}, respectively. As such, Z_3^{Min} and Z_3^{Max} are the lower and upper bounds of the third objective function, respectively. To find these bounds, we solve the model separately for each objective function. If only the makespan criterion is optimized, the objective values are $Z_1^* = 83, Z_2 = 1.60E + 08$ and $Z_3 = 22.9$. The CPU time for this run is 4.38 seconds. If only the environmental criteria are minimized, the objective values are $Z_1 = 90, Z_2^* = 1.26E + 08$ and $Z_3 = 23.9$. The CPU time for this run is 4.57 seconds. Finally, if the social criteria are maximized, the objectives are $Z_1 = 90, Z_2 = 1.51E + 08$ and $Z_3^* = 27.9$. The CPU time for this run is 4.27 seconds.

Therefore, the lower and upper bounds of the second objective are set to $Z_2^{Min} = 1.26E + 08$ and $Z_2^{Max} = 1.60E + 08$, respectively. Similarly, the lower and upper bounds of the third objective are respectively $Z_3^{Min} = 22.9$ and $Z_3^{Max} = 27.9$. To generate more Pareto solutions, the average of upper and lower bounds of the objectives is considered. However, there is no feasible solution when this average value is used. At the end, the total time to run the EC method to solve our industrial test, is 13.22 seconds.

All the Pareto solutions found by EC, SEO and LSEO are reported in Table 7. These solutions are depicted in Figure 6. One disadvantage of the EC method is that it is limited in its capacity to generate many Pareto solutions. However, SEO and LSEO are able to create 12 and 16 solutions, respectively. The results shown in Table 7 and Figure 6 confirm that SEO and LSEO are able to create high quality solutions like EC does.

Table 7. Pareto solutions after solving the industrial example

EC			SEO			LSEO		
Z_1	Z_2	Z_3	Z_1	Z_2	Z_3	Z_1	Z_2	Z_3
83	1.60E+08	22.9	85	1.85E+08	21.2	84	1.85E+08	20.9
90	1.26E+08	23.9	85	1.82E+08	21.4	84	1.83E+08	21.2
90	1.51E+08	27.9	86	1.80E+08	21.8	84	1.80E+08	21.8
-	-	-	87	1.78E+08	22.2	86	1.74E+08	22.5
-	-	-	87	1.76E+08	22.4	86	1.66E+08	23.2
-	-	-	88	1.72E+08	22.8	86	1.64E+08	23.6
-	-	-	88	1.66E+08	23.2	86	1.58E+08	24.2
-	-	-	88	1.64E+08	23.6	87	1.54E+08	24.6
-	-	-	89	1.58E+08	23.8	87	1.52E+08	25.2
-	-	-	89	1.54E+08	24.6	87	1.51E+08	25.5
-	-	-	89	1.52E+08	25.2	87	1.50E+08	25.8
-	-	-	90	1.49E+08	26.7	88	1.50E+08	26.2
-	-	-	-	-	-	88	1.49E+08	26.7
-	-	-	-	-	-	88	1.46E+08	26.9
-	-	-	-	-	-	88	1.45E+08	27.4
-	-	-	-	-	-	88	1.44E+08	27.6

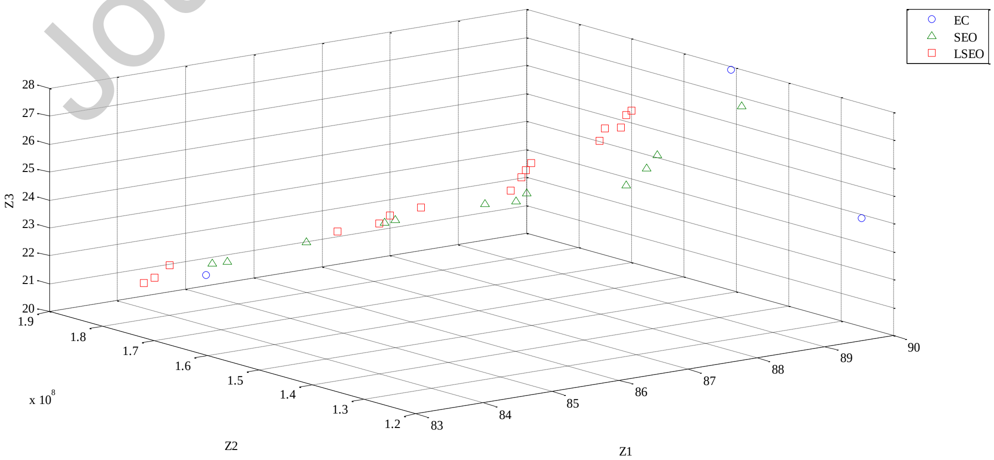
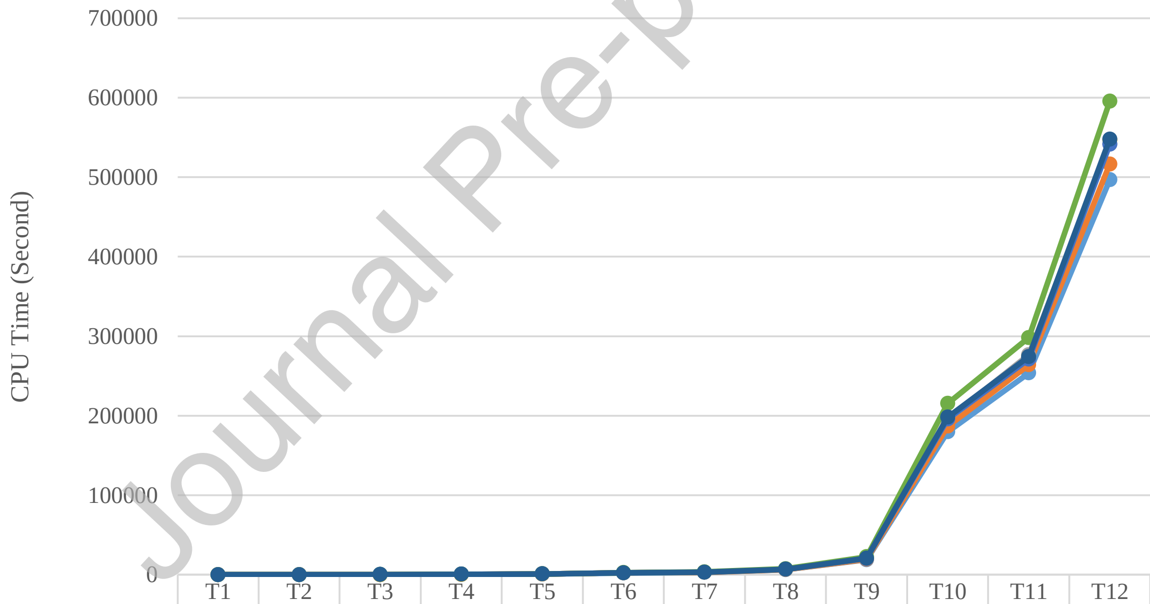


Figure 6. Pareto solutions for CWP company

5.4.Comparison

To show the high performance of the proposed LSEO, it has been compared to its original version of SEO as well as state-of-the-art methods like NSGA-II, SPEA2, and MOEA/D and two recent algorithms including MOBSO and MOKA. In this regard, 12 test problems with different complexity levels are solved by these algorithms. Due to the randomization of these algorithms, we run them thirty times and the average of their results is considered reliable.

The first criterion used in this comparison is the CPU time. This criterion also confirms the level of complexity of the test studies. Figure 7 shows the CPU times required by the algorithms to solve the simulated test studies. As can be seen, LSEO and SEO are faster than other algorithms. However, as can be seen from this chart, the CPU times of the algorithms are of the same order of magnitude. This is because the number of fitness evaluations is considered the same for all algorithms. From these results, SEO and MOBSO methods are identified as the fastest and slowest in the majority of the simulated test studies, respectively.



	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
SEO	15.163	21.412	99.7285	437.532	871.81	2228.66	2919.42	6319.98	18770.3	179625	253760	496634
LSEO	15.3146	21.6261	100.726	441.908	897.964	2295.52	3007	6509.58	19333.4	186810	263911	516499
NSGA-II	18.1956	25.6944	119.674	525.039	1020.02	2406.95	3152.98	6825.58	20459.6	195792	276599	541331
SPEA2	17.8923	25.2662	117.68	516.288	1009.82	2382.88	3184.5	6893.83	20664.2	197750	273833	546744
MOEA/D	20.4701	28.9062	134.633	590.669	999.719	2359.05	3152.66	6824.89	20457.6	195772	271095	541277
MOBSO	22.5171	31.7968	148.097	649.735	1099.69	2594.96	3467.93	7507.38	22503.4	215349	298204	595404
MOKA	20.7157	29.2531	136.249	597.757	1011.72	2387.36	3190.49	6906.79	20703.1	198121	274348	547772

Figure 7. CPU times of the algorithms for solving the simulated test studies

Four multi-objective criteria including NPS, MID, MS and HV are considered to evaluate the quality of Pareto solutions found by the algorithms. Their results are respectively reported in Table 8, 9, 10 and 11. In these tables, the best values are highlighted in bold.

Table 8. Evaluation of algorithm performance using the NPS metric

Test problem	SEO	LSEO	NSGA-II	SPEA2	MOEA/D	MOBSO	MOKA
T1	**14**	8	10	8	6	4	3
T2	20	**34**	26	16	15	8	12
T3	64	55	**76**	39	24	28	33
T4	23	**29**	16	8	19	18	15
T5	44	**62**	55	39	44	21	27
T6	76	78	**88**	56	74	36	25
T7	88	**102**	100	75	66	49	52
T8	105	**116**	100	79	81	68	42
T9	**215**	148	100	100	97	73	75
T10	**309**	188	100	100	100	93	56
T11	118	**172**	100	95	92	82	96
T12	136	**211**	100	100	100	100	88

Table 9. Evaluation of algorithms performance using the MID metric

Test problem	SEO	LSEO	NSGA-II	SPEA2	MOEA/D	MOBSO	MOKA
T1	39.6	40.8	33.2	29.8	37.5	26.5	**24.5**
T2	45.7	56.4	**29.8**	33.4	45.6	37.9	45.4
T3	102.6	88.7	78.6	92.7	**67.5**	85.2	86.5
T4	115.4	**95.4**	102.6	122.6	109.5	142.6	108.5
T5	276.3	188.7	**96.5**	112.5	119.6	189.4	242.5
T6	197.5	186.5	254.3	297.3	109.6	**98.3**	144.2
T7	**149.2**	156.2	188.7	206.3	188.5	193.2	174.5
T8	**228.4**	256.3	345.2	305.2	288.1	306.5	283.5
T9	319.5	**252.6**	428.9	402.4	377.5	392.6	275.1
T10	177.6	**144.3**	388.1	265.1	244.2	271.9	193.2
T11	188.9	**156.2**	504.2	209.3	235.1	228.3	298.2
T12	244.3	219.5	399.1	275.1	218.5	275.3	**199.4**

Table 10. Evaluation of algorithms performance using the MS metric

Test problem	SEO	LSEO	NSGA-II	SPEA2	MOEA/D	MOBSO	MOKA
T1	6984999.6	5894376.2	**7068555.1**	4982399.4	5068822.3	3894506.3	5884382.5
T2	6985734.5	**8648332.6**	5068429.5	2285694.1	6093392.5	8260555.1	7489302.5
T3	8443506.2	7085439.6	6089427.4	7053277.3	3885467.2	**8543772.8**	7094463.2
T4	**9956309.4**	8544288.3	7095275.2	4096855.3	4035588.2	3095668.2	6435068.2
T5	**10753982.3**	9885063.5	7068329.5	7047783.2	8490355.2	6988543.2	8665447.2
T6	8975664.2	**10546783.2**	8946684.5	8996583.2	6047752.5	7864733.5	9627543.5
T7	12527709.2	10546782.2	1078822.5	**12780423.2**	10247685.2	9987402.4	9902545.2
T8	**12563902.7**	12036547.3	11664893.4	10522910.3	9654553.6	10454893.2	11784405.2
T9	**13829504.2**	10863892.5	10573902.6	11829044.6	8562981.5	9924893.6	10452855.3
T10	12872895.3	11678649.3	11539671.5	10997783.5	10782861.5	11653782.5	**12994673.9**
T11	11673785.6	**12982901.5**	12738594.5	12452895.1	11770839.6	12620256.4	10097855.3

T12	13678864.7	**14014582.6**	12922099.1	12672895.1	11852856.4	10451197.5	10735784.6

Table 11. Evaluation of algorithms performance using the HV metric

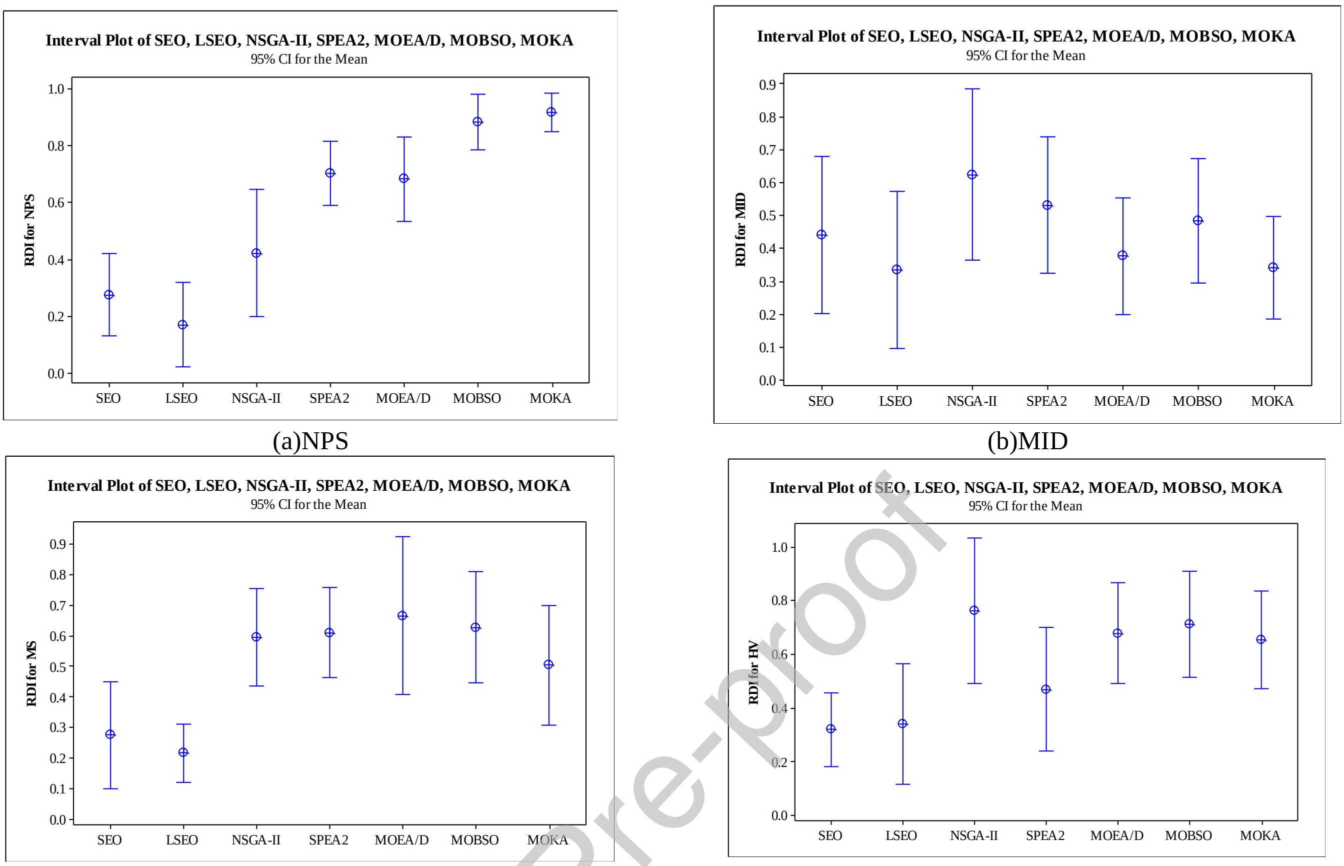
Test problem	SEO	LSEO	NSGA-II	SPEA2	MOEA/D	MOBSO	MOKA
T1	2.87E+09	**3.83E+09**	1.98E+09	3.72E+09	1.09E+09	1.54E+09	2.65E+09
T2	4.74E+09	5.53E+09	3.67E+09	**7.73E+09**	2.87E+09	1.93E+09	3.54E+09
T3	4.38E+09	**5.25E+09**	3.87E+09	4.29E+09	3.28E+09	2.99E+09	3.71E+09
T4	6.39E+09	7.8E+09	**8.54E+09**	6.85E+09	5.78E+09	4.18E+09	7.39E+09
T5	3.86E+09	**8.88E+09**	6.78E+09	7.38E+09	5.68E+09	7.58E+09	6.98E+09
T6	7.63E+09	7.98E+09	7.58E+09	6.58E+09	**9.38E+09**	6.18E+09	8.28E+09
T7	9.23E+09	**9.98E+09**	7.48E+09	8.38E+09	7.48E+09	8.38E+09	5.98E+09
T8	**1.86E+10**	1.15E+10	9.54E+09	1.04E+10	9.73E+09	1.82E+10	9.37E+09
T9	2.75E+10	1.86E+10	1.45E+10	2.67E+10	1.86E+10	**2.99E+10**	1.09E+10
T10	3.54E+10	**3.97E+10**	2.87E+10	1.87E+10	2.07E+10	2.18E+10	2.06E+10
T11	3.87E+10	2.58E+10	1.96E+10	**5.84E+10**	4.38E+10	2.97E+10	3.6E+10
T12	**6.85E+10**	3.29E+10	3.65E+10	4.87E+10	6.64E+10	3.79E+10	4.19E+10

To find the best algorithm, the Relative Deviation Index (RDI) [63] is used:

$$RDI = \frac{|Best_{Metric} - Alg_{Sol}|}{Max_{Sol} - Min_{Sol}} \tag{28}$$

where Max_{Sol} and Min_{Sol} are the maximum and minimum values for each metric, respectively. Alg_{Sol} is the value of a metric for a specific algorithm while $Best_{Metric}$ is the best value of the metric obtained among all the algorithms. It goes without saying that a lower value of RDI is more preferable.

After transforming the metrics based on the RDI, the interval plot for each metric is depicted statistically in Figure 8. Based on the NPS metric criterion (Figure 8(a)), the developed LSEO shows the best performance followed by the SEO algorithm. However, MOBSO and MOKA perform very poorly on this metric. Regarding the MID metric criterion (Figure 8(b)), again, the LSEO algorithm shows the best performance. MOKA is better than other algorithms in this case. The MOEA/D is also good on this criterion and better than the SEO algorithm. However, NSGA-II is the worst algorithm in this metric. Based on the MS metric (Figure 8(c)), SEO and LSEO algorithms are clearly better than other algorithms. For the HV metric (Figure 8(d)), the same conclusion is drawn from the results. In conclusion, as can be seen from the interval plots, the proposed LSEO algorithm achieves the best performance in this comparative study.



(a)NPS (b)MID

(c)MS (d)HV

Figure 8. Interval plots based on RDI analyzing the performance of the algorithms in each metric: (a)NPS, (b)MID, (c)MS and (d)HV

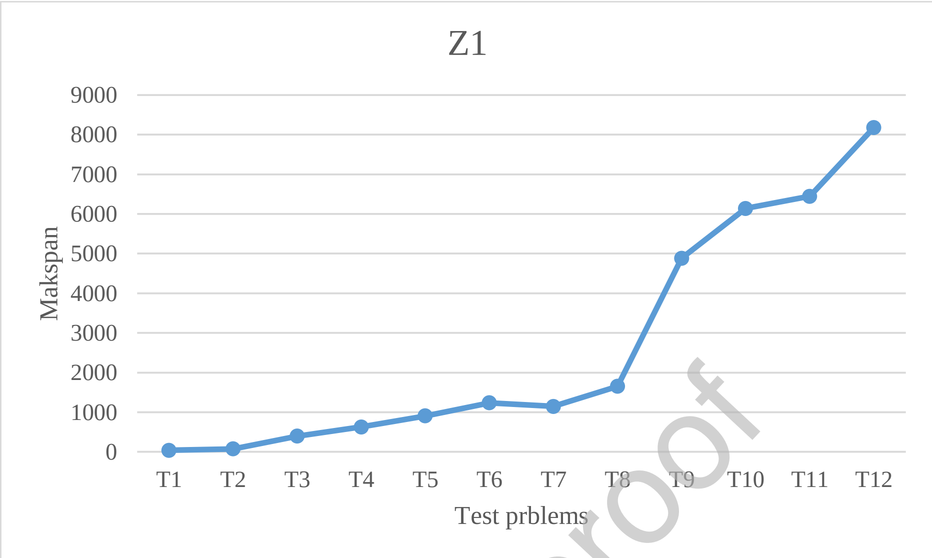
5.5.Sensitivity analyses

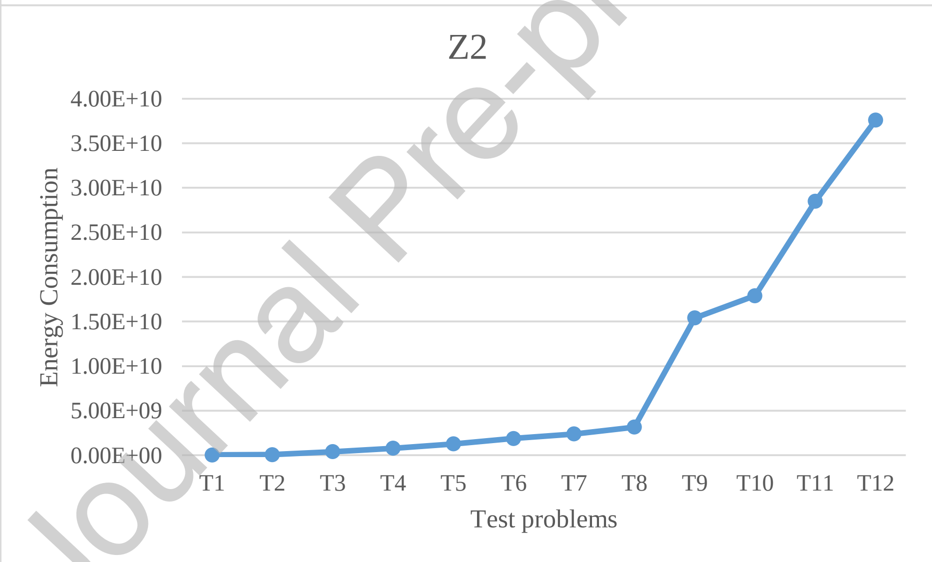
To evaluate the robustness of the optimization model developed, some sensitivity studies are carried out here. First, the Pareto solutions found by LSEO to be the best algorithm in this study, are sorted by the ideal distance criterion. Then, the first solution of this Pareto set is noted in Table 12. The variations of the objectives in this table are shown in Figure 9.

Table 12. The first solution from the set of sorted Pareto solutions from LSEO

Test problem	Z_1	Z_2	Z_3
T1	39	3,30E+07	10
T2	76	7,78E+07	10,9
T3	399	4,07E+08	19,6
T4	631	7,87E+08	41,2
T5	909	1,27E+09	65,9
T6	1236	1,89E+09	47,9
T7	1146	2,40E+09	78,9
T8	1655	3,16E+09	117,9
T9	4885	1,54E+10	208,9
T10	6141	1,79E+10	192,1

T11	6446	2,85E+10	355,8
T12	8177	3,76E+10	487,1





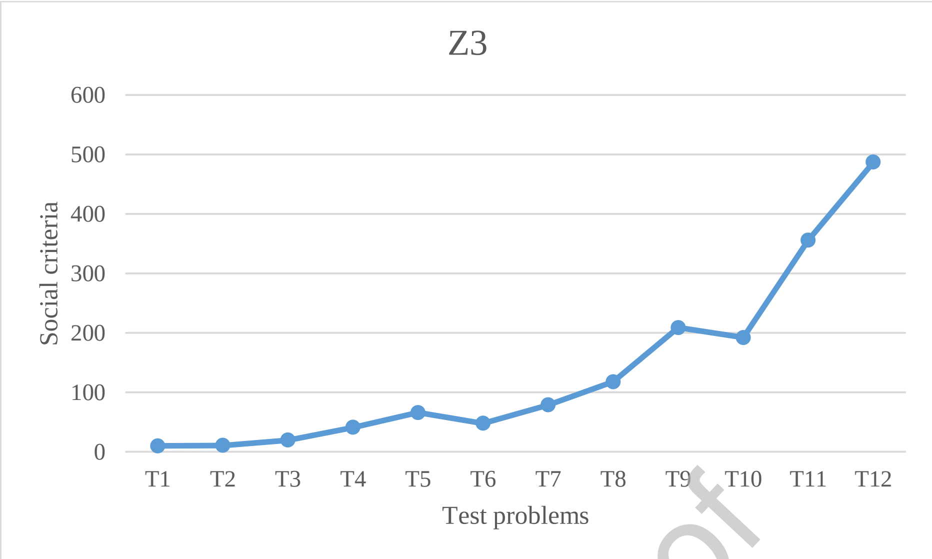


Figure 9. Variations of the objective values

To show the impact of the parameters of the model developed for the decision-makers of the CWP company as a leader in the wood industry in Canada, the EC method is selected to solve the industrial example in our sensitivity analyses. Of these parameters, four important ones are selected for modification. The company's budget (B), the maximum waste (MW) as well as the social weights for job opportunities (WJ) and lost working days (WL) are considered for our analyses. For each parameter, certain modifications are done by four scenarios: S1 to S4 and the values of objective functions in each scenario are indicated. Note that in our sensitivity analyses, in addition to makespan, the total flow-time ($\sum_{f=1}^{F} CT_f$) is also considered to better show the impact of parameters on the economic criteria.

The sensitivity analysis of the company's budget is reported in Table 13. In four scenarios, the company's budget goes from 2594561$ (S1) to 2000000$ (S4). The values of the objective functions for each case are noted. As can be seen, there is no change in the makespan criterion, considered as the first objective while an increase in the total flow-time is observed in the last scenario. This means that reducing the budget to be less than 2300000$ has a negative economic impact. Although the values of the second and third objectives have some variations, they have been increased if the first scenario is compared to the last scenario. The behavior of these objectives (except makespan) is drawn in Figure 10.

Table 13. Sensitivity analysis on the budget of company

Scenarios	Value of the company budget ($)	Z_1 (Makespan)	Z_1 (Flow-time)	Z_2	Z_3
S1	2594561	83	231	1.60E+08	22.9
S2	2400000	83	231	1.39E+08	24.2
S3	2300000	83	231	1.51E+08	25
S4	2000000	83	235	1.73E+08	24.2

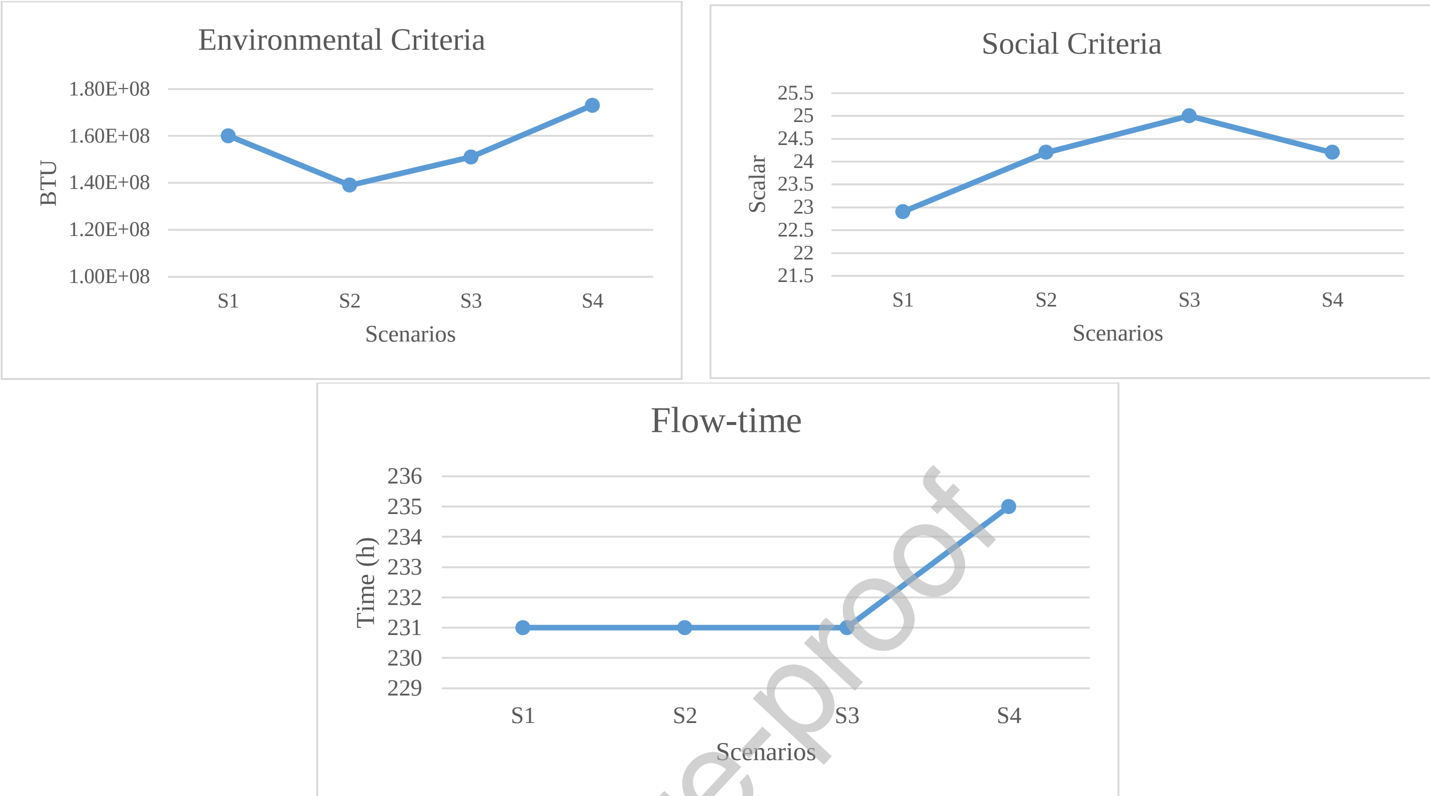


Figure 10. Sensitivity analysis on the budget of company

As indicated in Table 14, the sensitivity analysis is performed on the value of the maximum waste products. We have reduced the maximum waste products from 2.5 units to 1 unit. This parameter only has an impact on the values of the environmental criteria. The economic criteria including both makespan and total flow-time as well as the third objective functions have not changed. Decreasing the maximum amount of waste products leads to an increase in energy consumption as a second objective. This behavior is illustrated in Figure 11.

Table 14. Sensitivity analysis on the maximum waste products

Number of Scenario	Value of maximum waste products	Z_1 (Makespan)	Z_1 (Flow-time)	Z_2	Z_3
S1	2.5	83	231	1.53E+08	22.9
S2	2	83	231	1.60E+08	22.9
S3	1.5	83	231	1.71E+08	22.9
S4	1	83	231	1.74e+08	22.9

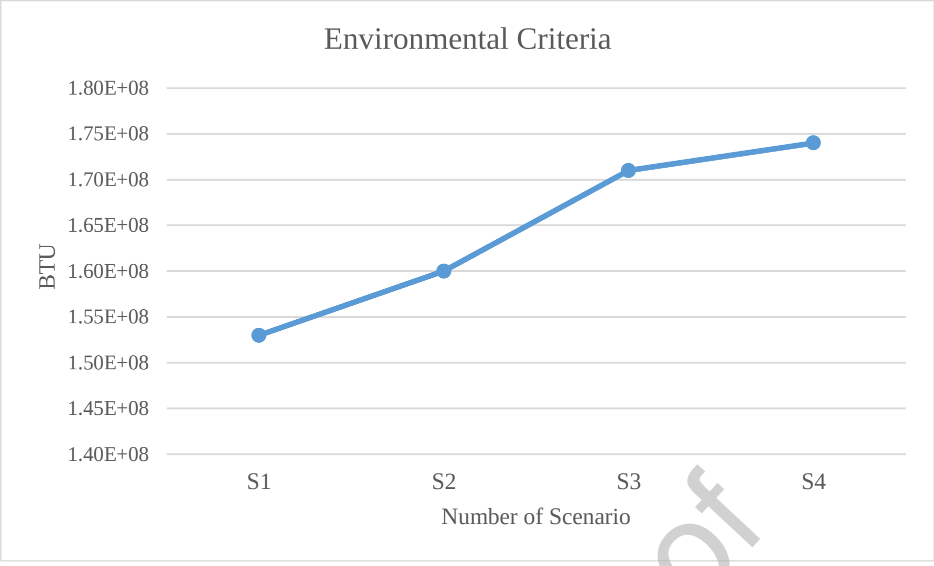
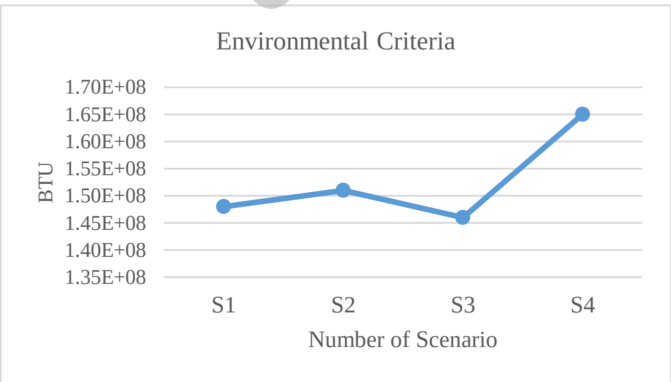


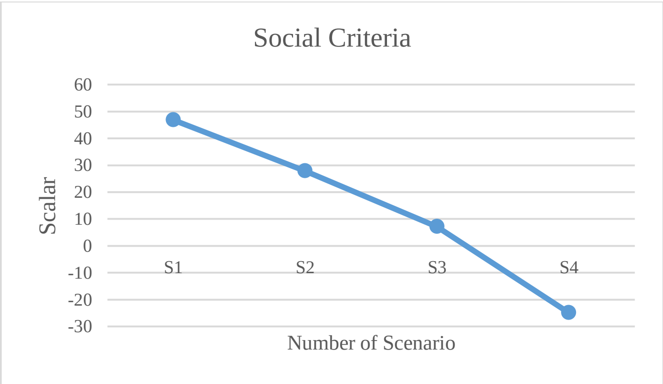
Figure 11. Sensitivity analysis on the maximum waste products

The final sensitivity analysis looks at the value of social weights for job opportunities and lost working days as the third objective function as shown in Table 15. To better show the impact of changes on these two parameters, the main objective of the EC is changed from makespan to social criteria as the third objective function. In the four scenarios, we reduced the impact of job opportunities while increasing the impact of lost working days. Except for makespan, other criteria show changes. These behaviors are illustrated in Figure 12. It shows that the total flow-time decreases in these scenarios except in S2. The energy consumption shows an increase if the first scenario is compared to the last scenario. However, in the S3 scenario, it shows a reduction. Finally, the social objective shows a strong reduction in all scenarios.

Table 15. Sensitivity analysis on the social weights

Number of Scenario	Social weights	Z_1 (Makespan)	Z_1 (Flow-time)	Z_2	Z_3
S1	*WJ*=0.99; *WL*=0.01;	90	238	1.48E+08	46.89
S2	*WJ*=0.9; *WL*=0.1;	90	238	1.51E+08	27.9
S3	*WJ*=0.8; *WL*=0.2;	90	235	1.46E+08	7.2
S4	*WJ*=0.7; *WL*=0.3;	90	233	1.65E+08	-24.9





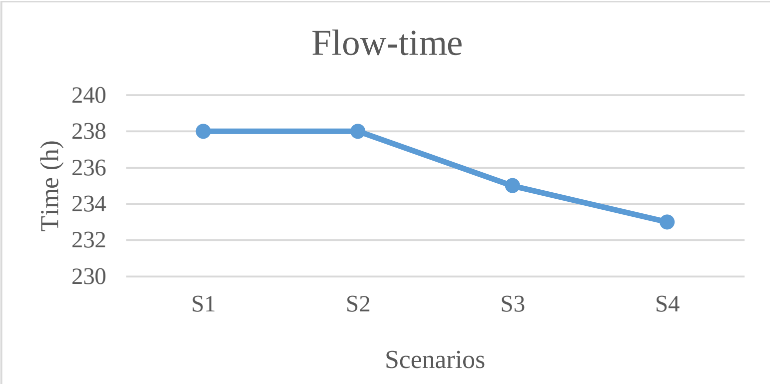


Figure 12. Sensitivity analysis on the social weights

6. Conclusion, recommendations and further research

The DPFSP traditionally aims to minimize makespan or total flow time based on economic criteria. However, based on the concept of TBL, traditional modeling of the DPFSP is not able to simultaneously cover all economic, environmental and social criteria. This study developed a sustainable DPFSP with the assumption of different production centers and technologies on machines that have a strong impact on environmental and social criteria. This study which considers job opportunities and lost working days as social factors is the first study in the area of DPFSP. Therefore, a multi-objective optimization model was developed to approach a sustainable TBL-based DPFSP.

One idea of this paper was to virtually meet the challenge of sustainable development based on the TBL concept for wood production in Canada. In this regard, CWP has been selected as a full-scale application for our optimization model. Having different simulated test studies to analyze the complexity of this NP-hard model, this study proposed a multi-objective learning-based heuristic called LSEO and compared it to several recent and state-of-the art algorithms from the literature.

The results show the viability of the proposed sustainable DPFSP. First of all, the feasibility of the developed optimization model has been shown by a numerical example as given in Figure 1. The optimal Pareto solutions for solving the case of the company CWP have been shown in Figure 6 to confirm the optimality of our solutions compared to the exact solver using the EC method. The high performance of the proposed LSEO was shown in different criteria (Figures 7 and 8) to confirm its superiority over other algorithms. The variations of the sustainability objectives are illustrated in Figure 9. Finally, the efficiency of the optimization model developed was analyzed by certain sensitivity analyses as indicated in Figures 10 to 12.

From the results, some recommendations can be suggested. First, this study conceptually shifts the energy-efficient DPFSP to the sustainable DPFSP to simultaneously cover all the economic, environmental and social factors. The use of different production technologies can be defined as an introduction to the reverse production and supply chains with multiple production centers. A high number of Pareto solutions found by algorithms gives production managers this possibility to find an interaction between economic, environmental and social alternatives. Last but not least, setting the parameters of the model such as the company's budget or the social weights, is very important to achieve the environmental and social sustainability for a production system. In the continuation of this work, we will try to obtain data from an actual industrial case study, and develop simple and accessible guidelines to help production managers to implement the concept of triple bottom line for production systems.

In conclusion, although this study is more complex than the majority of existing papers in the area of DPFSP, there are many suggestions to continue this line of research as follows:

- Uncertain factors in the definition of DPFSP may be used. The use of robust and stochastic optimization concepts can be suggested to resolve the uncertainty.
- Adding risk factors based on economic, environmental and social criteria to the DPFSP is rarely considered and can be suggested.
- The application of the proposed algorithm to other combinatorial optimization problems such as home healthcare systems and facility location planning, as well as the development of this method with more learning and local search techniques, are some of the potential continuations of this article.

Credit Author Statement:

Amir M. Fathollahi-Fard: Conceptualization; Formal analysis; Investigation; Methodology; Software; Validation; Original draft; Visualization; Review & Editing;

Lyne Woodward: Supervision; Project Admiration; Review & Editing;

Ouassima Akhrif: Supervision; Review & Editing;

Acknowledgement:

The authors declare that they have NO affiliation with or involvement in any organization or entity such as Canadian Wood Products (CWP) with financial interest or non-financial interest such as personal or professional relationships, affiliations, knowledge or benefits in the method and materials discussed in this paper. This work was supported by the discovery grant program from the National Sciences and Engineering Research Council of Canada (NSERC), grant number RGPIN-2019-05853.

References:

[1] Fathollahi-Fard, A. M., Ahmadi, A., & Mirzapour Al-e-Hashem, S. M. J., (2020). Sustainable Closed-loop Supply Chain Network for an Integrated Water Supply and Wastewater Collection System under Uncertainty, *Journal of Environmental Management*, 275, 111277.

[2] Lu, C., Gao, L., Gong, W., Hu, C., Yan, X., & Li, X. (2020). Sustainable Scheduling of Distributed Permutation Flow-shop with Non-identical Factory Using A Knowledge-based Multi-objective Memetic Optimization Algorithm. *Swarm and Evolutionary Computation*, 100803.

[3] Shao, W., Shao, Z., & Pi, D. (2020). Modeling and multi-neighborhood iterated greedy algorithm for distributed hybrid flow shop scheduling problem. *Knowledge-Based Systems*, 105527.

[4] Han, Y., Li, J., Sang, H., Liu, Y., Gao, K., & Pan, Q. (2020). Discrete evolutionary multi-objective optimization for energy-efficient blocking flow shop scheduling with setup time. *Applied Soft Computing*, 106343.

[5] Mansouri, S. A., Aktas, E., & Besikci, U. (2016). Green scheduling of a two-machine flowshop: trade-off between makespan and energy consumption. *European Journal of Operational Research*, 248(3), 772-788.

[6] Xu, Y., Yan, C., Liu, H., Wang, J., Yang, Z., & Jiang, Y. (2020). Smart Energy Systems: A Critical Review on Design and Operation Optimization. *Sustainable Cities and Society*, 102369.

[7] Lu, C., Gao, L., Li, X., & Chen, P. (2016). Energy-efficient multi-pass turning operation using multi-objective backtracking search algorithm. *Journal of Cleaner Production*, 137, 1516-1531.

[8] Wang, G., Gao, L., Li, X., Li, P., & Tasgetiren, M. F. (2020). Energy-efficient distributed permutation flow shop scheduling problem using a multi-objective whale swarm algorithm. *Swarm and Evolutionary Computation*, 100716.

[9] Ye, H., Wang, X., & Liu, K. (2020). Adaptive Preventive Maintenance for Flow Shop Scheduling With Resumable Processing. *IEEE Transactions on Automation Science and Engineering*. 10.1109/TASE.2020.2978890

[10] Zhu, G. Y., Ding, C., & Zhang, W. B. (2020). Optimal foraging algorithm that incorporates fuzzy relative entropy for solving many-objective permutation flow shop scheduling problems. *IEEE Transactions on Fuzzy Systems*. 28(11), 2738 – 2746.

[11] Duarte, J. L. R., Fan, N., & Jin, T. (2020). Multi-process production scheduling with variable renewable integration and demand response. *European Journal of Operational Research*, 281(1), 186-200.

[12] Anghinolfi, D., Paolucci, M., & Ronco, R. (2020). A bi-objective heuristic approach for green identical parallel machine scheduling. *European Journal of Operational Research*, *289*(2), 416-434.

[13] Abreu, L. R., Cunha, J. O., Prata, B. A., & Framinan, J. M. (2020). A genetic algorithm for scheduling open shops with sequence-dependent setup times. *Computers & Operations Research*, 113, 104793.

[14] Ghaleb, M., Zolfagharinia, H., & Taghipour, S. (2020). Real-time production scheduling in the Industry-4.0 context: Addressing uncertainties in job arrivals and machines breakdowns. *Computers & Operations Research*, 105031.

[15] Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, *1*(1), 67-82.

[16] Fathollahi-Fard, A. M., Hajiaghaei-Keshteli, M., & Tavakkoli-Moghaddam, R. (2018). The social engineering optimizer (SEO). *Engineering Applications of Artificial Intelligence*, *72*, 267-293.

[17] Naderi, B., & Ruiz, R. (2010). The distributed permutation flowshop scheduling problem. *Computers & Operations Research*, *37*(4), 754-768.

[18] Gao, J., & Chen, R. (2011). A hybrid genetic algorithm for the distributed permutation flowshop scheduling problem. *International Journal of Computational Intelligence Systems*, *4*(4), 497-508.

[19] Lin, S. W., Ying, K. C., & Huang, C. Y. (2013). Minimising makespan in distributed permutation flowshops using a modified iterated greedy algorithm. *International Journal of Production Research*, *51*(16), 5029-5038.

[20] Naderi, B., & Ruiz, R. (2014). A scatter search algorithm for the distributed permutation flowshop scheduling problem. *European Journal of Operational Research*, *239*(2), 323-334.

[21] Bargaoui, H., Driss, O. B., & Ghédira, K. (2017). A novel chemical reaction optimization for the distributed permutation flowshop scheduling problem with makespan criterion. *Computers & Industrial Engineering*, *111*, 239-250.

[22] Fernandez-Viagas, V., Perez-Gonzalez, P., & Framinan, J. M. (2018). The distributed permutation flow shop to minimise the total flowtime. *Computers & Industrial Engineering*, *118*, 464-477.

[23] Pan, Q. K., Gao, L., Wang, L., Liang, J., & Li, X. Y. (2019). Effective heuristics and metaheuristics to minimize total flowtime for the distributed permutation flowshop problem. *Expert Systems with Applications*, *124*, 309-324.

[24] Ruiz, R., Pan, Q. K., & Naderi, B. (2019). Iterated Greedy methods for the distributed permutation flowshop scheduling problem. *Omega*, *83*, 213-222.

[25] Meng, T., Pan, Q. K., & Wang, L. (2019). A distributed permutation flowshop scheduling problem with the customer order constraint. *Knowledge-Based Systems*, *184*, 104894.

[26] Wang, J. J., & Wang, L. (2018). A knowledge-based cooperative algorithm for energy-efficient scheduling of distributed flow-shop. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, (99), 1-15.

[27] Fu, Y., Tian, G., Fathollahi-Fard, A. M., Ahmadi, A., & Zhang, C. (2019). Stochastic multi-objective modelling and optimization of an energy-conscious distributed permutation flow shop scheduling problem with the total tardiness constraint. *Journal of cleaner production*, *226*, 515-525.

[28] Xu, Y., Wang, L., Wang, S., & Liu, M. (2014). An effective hybrid immune algorithm for solving the distributed permutation flow-shop scheduling problem. *Engineering Optimization*, *46*(9), 1269-1283.

[29] Jing, X. L., Pan, Q. K., & Gao, L. (2021). Local search-based metaheuristics for the robust distributed permutation flowshop problem. *Applied Soft Computing*, 107247.

[30] Huang, J., & Gu, X. (2021). Distributed assembly permutation flow-shop scheduling problem with sequence-dependent set-up times using a novel biogeography-based optimization algorithm. *Engineering Optimization*, 1-21.

[31] Sabuj, S. U., Ali, S. M., Hasan, K. W., & Paul, S. K. (2021). Contextual relationships among key factors related to environmental sustainability: Evidence from an emerging economy. *Sustainable Production and Consumption*, *27*, 86-99.

[32] Karmaker, C. L., Ahmed, T., Ahmed, S., Ali, S. M., Moktadir, M. A., & Kabir, G. (2020). Improving supply chain sustainability in the context of COVID-19 pandemic in an emerging economy: Exploring drivers using an integrated model. *Sustainable production and consumption*, *26*, 411-427.

[33] Mojtahedi, M., Fathollahi-Fard, A. M., Tavakkoli-Moghaddam, R., & Newton, S., (2021). Sustainable vehicle routing problem for coordinated solid waste management. *Journal of Industrial Information Integration.* https://doi.org/10.1016/j.jii.2021.100220

[34] Corbett, C. J., & Kirsch, D. A. (2001). International diffusion of ISO 14000 certification. *Production and operations management*, *10*(3), 327-342.

[35] Benoît, C., Norris, G. A., Valdivia, S., Ciroth, A., Moberg, A., Bos, U., ... & Beck, T. (2010). The guidelines for social life cycle assessment of products: just in time! *The international journal of life cycle assessment*, *15*(2), 156-163.

[36] Llach, J., Marimon, F., & del Mar Alonso-Almeida, M. (2015). Social Accountability 8000 standard certification: analysis of worldwide diffusion. *Journal of Cleaner Production*, *93*, 288-298.

[37] Ali, S. M., Paul, S. K., Chowdhury, P., Agarwal, R., Fathollahi-Fard, A. M., Jabbour, C. J. C., & Luthra, S. (2021). Modelling of supply chain disruption analytics using an integrated approach: An emerging economy example. *Expert Systems with Applications*, 114690.

[38] Marimuthu, R., Sankaranarayanan, B., Ali, S. M., de Sousa Jabbour, A. B. L., & Karuppiah, K. (2021). Assessment of key socio-economic and environmental challenges in the mining industry: Implications for resource policies in emerging economies. *Sustainable Production and Consumption*. 27, 814-830.

[39] Fathollahi-Fard, A. M., Ranjbar-Bourani, M., Cheikhrouhou, N., & Hajiaghaei-Keshteli, M. (2019). Novel modifications of social engineering optimizer to solve a truck scheduling problem in a cross-docking system. *Computers & Industrial Engineering*, *137*, 106103.

[40] Dulebenets, M. A. (2020). An Adaptive Island Evolutionary Algorithm for the berth scheduling problem. *Memetic Computing*, *12*(1), 51-72.

[41] Pasha, J., Dulebenets, M. A., Kavoosi, M., Abioye, O. F., Wang, H., & Guo, W. (2020). An Optimization Model and Solution Algorithms for the Vehicle Routing Problem With a “Factory-in-a-Box”. *IEEE Access*, *8*, 134743-134763.

[42] Gonçalves, J. F., & Resende, M. G. (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, *17*(5), 487-525.

[43] Dulebenets, M. A. (2021). An Adaptive Polyploid Memetic Algorithm for Scheduling Trucks at a Cross-Docking Terminal. *Information Sciences*. https://doi.org/10.1016/j.ins.2021.02.039

[44] Fathollahi-Fard, A. M., Ahmadi, A., Goodarzian, F., & Cheikhrouhou, N. (2020). A bi-objective home healthcare routing and scheduling problem considering patients’ satisfaction in a fuzzy environment. *Applied soft computing*, 106385.

[45] Pasha, J., Dulebenets, M. A., Kavoosi, M., Abioye, O. F., Theophilus, O., Wang, H., ... & Guo, W. (2020). Holistic tactical-level planning in liner shipping: an exact optimization approach. *Journal of Shipping and Trade*, *5*, 1-35.

[46] Zhang, B., Pan, Q. K., Gao, L., Li, X. Y., Meng, L. L., & Peng, K. K. (2019). A multiobjective evolutionary algorithm based on decomposition for hybrid flowshop green scheduling problem. *Computers & Industrial Engineering*, *136*, 325-344.

[47] Zuo, M., Dai, G., Peng, L., Wang, M., Liu, Z., & Chen, C. (2020). A case learning-based differential evolution algorithm for global optimization of interplanetary trajectory design. *Applied Soft Computing*, *94*, 106451.

[48] Qu, C., Gai, W., Zhong, M., & Zhang, J. (2020). A novel reinforcement learning based grey wolf optimizer algorithm for unmanned aerial vehicles (UAVs) path planning. *Applied Soft Computing*, *89*, 106099.

[49] Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *International conference on parallel problem solving from nature* (pp. 849-858). Springer, Berlin, Heidelberg.

[50] Zitzler, Eckart, LAUMANNS, Marco, & Thiele, Lothar (2001). SPEA2: Improving the strength Pareto evolutionary algorithm. *TIK-report*, 103, 34-76.

[51] Zhang, Qingfu et Li, Hui. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 2007, 11(6), 712-731.

[52] Cheraghalipour, A., Paydar, M. M., & Hajiaghaei-Keshteli, M. (2018). A bi-objective optimization for citrus closed-loop supply chain using Pareto-based algorithms. *Applied Soft Computing*, *69*, 33-59.

[53] Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, *8*(2), 173-195.

[54] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., & Da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on evolutionary computation*, *7*(2), 117-132.

[55] Zitzler, E. (1999). *Evolutionary algorithms for multiobjective optimization: Methods and applications* (Vol. 63). Ithaca: Shaker.

[56] Zitzler, E., Brockhoff, D., & Thiele, L. (2007). The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 862-876). Springer, Berlin, Heidelberg.

[57] Zahedi, A., Salehi-Amiri, A., Hajiaghaei-Keshteli, M., & Diabat, A. (2021). Designing a closed-loop supply chain network considering multi-task sales agencies and multi-mode transportation. *Soft Computing*, 1-33.

[58] Hajiaghaei-Keshteli, M., & Sajadifar, S. M. (2010). Deriving the cost function for a class of three-echelon inventory system with N-retailers and one-for-one ordering policy. *The International Journal of Advanced Manufacturing Technology*, *50*(1-4), 343-351.

[59] Salehi-Amiri, A., Zahedi, A., Akbapour, N., & Hajiaghaei-Keshteli, M. (2021). Designing a sustainable closed-loop supply chain network for walnut industry. *Renewable and Sustainable Energy Reviews*, *141*, 110821.

[60] Hajiaghaei-Keshteli, M., Sajadifar, S. M., & Haji, R. (2011). Determination of the economical policy of a three-echelon inventory system with (R, Q) ordering policy and information sharing. *The International Journal of Advanced Manufacturing Technology*, *55*(5-8), 831-841.

[61] Roy, R. K. (2010). *A primer on the Taguchi method*. Society of Manufacturing Engineers.

[62] Haimes, Y.Y., Ladson, L.S., & Wismer, D.A., (1971). Bicriterion formulation of problems of integrated system identification and system optimization. *IEEE Transaction on Systems, Man and Cybernetic*. 296 (3), 34-41.

[63] Rothe, I., Susse, H., & Voss, K. (1996). The method of normalization to determine invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *18*(4), 366-376.