European Journal of Operational Research 307 (2023) 519–537

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

Energy-aware production scheduling in the flow shop environment under sequence-dependent setup times, group scheduling and renewable energy constraints

Masoumeh Ghorbanzadeh, Mohammad Ranjbar*

Department of Industrial Engineering, Faculty of Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

ARTICLE INFO

Article history:
Received 25 December 2021
Accepted 25 September 2022
Available online 29 September 2022

Keywords:
Scheduling
Sequence-dependent flow shop
Energy
Mixed-integer programming
Heuristic

ABSTRACT

In this paper, we investigate an energy-aware flow shop scheduling problem under sequence-dependent setup times, group scheduling, and renewable energy constraints. We aim to minimize the total energy cost dependent on time-of-use electricity tariffs. To this end, we develop two mixed-integer linear programming models, including a time-unit index model and a time-interval index model. Besides, we develop a decomposition-based heuristic algorithm to solve efficiently medium-size instances. Using extensive computational experiments, we show that the heuristic algorithm outperforms both developed models, and the time-interval index model indicates superior performance than the time-unit index model. Finally, we provide a set of sensitivity analyses and evaluation of economic performance.

1. Introduction

Most of the consumed energy in the world is in the form of electricity in the industrial sector (Zhang, 2016). Electricity is a form of energy that cannot be stored efficiently and should be distributed and consumed immediately after being produced. In other words, we need to keep a balance between electricity supply and demand. To do so, in many countries, time-of-use (TOU) pricing has been considered to encourage electricity consumers, mainly factories with high electricity consumption, to shift their electricity consumption from on-peak hours to mid-peak or off-peak hours (Zhang, 2016). The use of renewable energy is another policy to reduce electricity demand from the grid and prevent air pollution. On-site renewable energy production can be a valuable resource for companies where energy has a significant impact on their production processes. It will reduce their dependence on power plants and counteract surging electricity costs. In addition, the use of renewable energy can make a significant contribution to reducing the environmental impact of production processes. To reduce the cost of electricity and take advantage of all possible opportunities, it is necessary to develop optimal production schedules so that renewable energy can be used as much as possible, and if there is a need to buy electricity, the lowest cost is paid. In the contemporary era, the number of factories that use renewable energy is increasing every day. It is due to the high cost of purchasing electricity from the grid, as well as reducing air pollution and feed-in-tariff schemes that governments have put in place to use renewable energy. There are several companies that use on-site renewable energy, such as LG Gumi Factory, Brabantia, SIG Combibloc, and Whirlpool.

In this paper, we design two models and an algorithm to develop schedules in the flow shop environment. We develop our schedules based on group scheduling and sequence-dependent setup times assumptions because group scheduling has significant applications in various industries such as automobile, aerospace and semiconductor light industries and, as far as we know, has not already been addressed in the literature.

What motivated us to carry out this research is its applications in semiconductor light industries in which products such as LCD, TFT-LCD, LED and OLED are produced where each one is made of several components. For instance, Fig.1 presents the components of TFT-LCD panel.

Each component of TFT-LCD panel is produced in a flow shop environment in which group scheduling is used. Jobs are divided into groups based on their shapes, sizes and technological constraints. The jobs of each group may have different processing times and power consumption. For instance, in the polarizer attachment, which is one of the main steps of the polarizer film production, the jobs of each group usually have different processing times, and there are tiny setup times between jobs due to small changes in machines temperature. We ignore such setup times and we consider only setup times between groups that are noticeable. It should be noted that the number of semiconductor light factories that use on-site renewable energy is growing. For example, LG

* Corresponding author at: Ferdowsi University of Mashhad, Department of Industrial Engineering, Faculty of Engineering, No. 516, Mashhad, Iran.
E-mail address: m_ranjbar@um.ac.ir (M. Ranjbar).

https://doi.org/10.1016/j.ejor.2022.09.034
0377-2217/

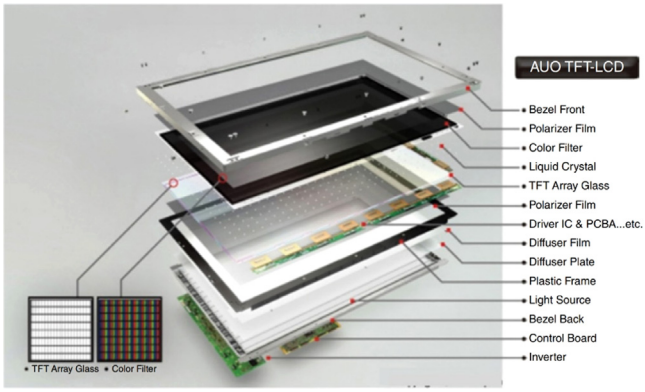

Fig. 1. The components of TFT-LCD panel (Shahvari, 2017).

Gumi Factory, which is a TV manufacturer in Korea uses solar panels in the factory to provide part of its electricity needs.

Most previous research has developed time-index models for similar problems causing large CPU runtimes. It should be mentioned that these models do not address group scheduling challenges. For instance, Schulz et al. (2019) used a time index model based on the minute to formulate the problem and assumed each processing time has a discrete uniform distribution taken from {1,...,10}. They showed that the average CPU runtime for a problem that includes 10 jobs is around 67 hours. Some other papers have considered time units more than one minute and have presumed that each processing time is a multiplication of the considered time unit. For example, Zhang et al. (2017) considered a 20-minute time unit and assumed that each processing time is a multiplication of 20 minutes. They considered a flexible flow shop scheduling problem including two jobs, two stages, and two machines in each stage and solved it in eight hours. Since scheduling problems, particularly the problem at hand, are a kind of short-term planning, they must be solved in a short CPU runtime. Moreover, processing times are not always a multiplication of a specific time unit, and hence we need to develop new models to cover previous deficiencies. To do so, we employ the concept of time interval to formulate the problem in a more efficient way. We develop two mixed-integer programming (MIP) models, one based on the time-unit index model and the other one based on the time-interval index model, to compare their performances. The former is developed based on the ideas that have already been in the literature whereas the latter involve new ideas. Furthermore, due to the fact that MIP models usually fail to solve medium-size instances, we develop a decomposition-based heuristic algorithm (DBHA) to cope with this problem.

The contributions of this research include the followings: (1) we develop two MIP models based on the time-unit index and the time-interval index ideas; (2) we develop a DBHA that outperforms developed models, especially for medium-size instances; and (3) we present a set of economic analyses for on-site renewable energy consumption.

The rest of this paper is organized as follows. The related research works are reviewed in Section 2. In Section 3, a detailed description of the problem is provided, and two MIP models are developed. In Section 4, a decomposition-based heuristic algorithm is designed to solve the problem. In Section 5, the developed solutions approaches are compared and evaluated. Finally, in Section 6, a summary and suggestions for future research are provided.

2. Related work

We categorize the related published research works into two parts. In the first part, we review the papers that investigate energy-aware scheduling in shop environments. In the second part, we review the research works that have studied group scheduling in the flow shop environment. In both categories, there are some papers that have considered sequence-dependent setup times.

2.1. Energy-aware scheduling in shop environments

Several papers presented in this field have addressed reducing consumed energy or abatement of consumed electricity cost by considering various solution approaches. For instance, some of these approaches are shutting down production machines in the idle state, reducing the production rate at on-peak hours, changing the speed of machines, and changing the sequence of jobs or their assignment to machines. Giret et al. (2015), Gahm et al. (2016), Biel and Glock (2016), as well as Akbar and Irohara (2018) are some review articles in this field. Several articles published in this field have only sought to reduce energy consumption without considering electricity costs at different hours of a day. For example, because shutting down machines in the idle state can significantly reduce the consumed energy, Mouzon et al. (2010) investigated this policy for machines that were not bottleneck. Tang et al. (2016) investigated the flexible flow shop scheduling problem by considering the possibility of changing the machines speed and the possibility of machines breakdown. Mansouri et al. (2016) developed a bi-objective model for minimizing energy consumption and makespan in a two-machine flow shop scheduling problem based on the idea that the speed increase of a machine might decrease job processing times and increase energy consumption. They assumed that sequence-dependent setup times between jobs, and they solved their developed model using a heuristic algorithm. Moreover, Li et al. (2018) presented a paper on the flexible flow shop environment with sequence-dependent setup times. Taking into account the energy consumed during processing, standby state, and setup times, they proposed a multi-objective algorithm to minimize the makespan and energy consumption. Wu and Sun (2018) proposed a multi-objective model and algorithm for the job shop scheduling problem. They presumed that shutting down a machine consumes less energy than leaving the machine idle when there is no job to process, and starting up and shutting down a machine too often will cause it to breakdown. They considered minimizing the makespan, the energy consumption, and the number of starting ups and shutting downs of machines as the objectives of their model.

Some researchers tried to reduce energy costs by taking into account the variable cost of electricity at different working hours. For example, considering electricity tariffs, Wang et al. (2017) presented a single-objective model for the two-machine permutation flow shop scheduling problem. They solved the problem using two heuristic algorithms based on Johnson's rule and dynamic programming. Furthermore, Ho et al. (2020) studied a two-machine flow shop scheduling problem to minimize the electricity cost. They developed a heuristic algorithm including two phases. In the first phase, a series of schedules with minimum makespan is obtained, and in the second phase, one of the schedules obtained in the first phase leading to minimum electric cost is selected using the shortest path algorithm. Schulz et al. (2019) modeled the flexible flow shop scheduling problem with the three objectives of minimizing the makespan, energy cost, and peak power consumption and used a local search algorithm to solve the problem. Moreover, Cui and Lu (2021) developed a single-objective model to minimize electricity cost in a flow shop environment under TOU electricity tariff. They captured the preventive maintenance operations and designed a two-layer mathheuristic to solve the problem. In the external layer, the sequence of jobs is determined using a genetic algorithm, and in the internal layer, the maintenance schedule is constructed through a dynamic programming algorithm.

Some research studies have also considered renewable energy and have investigated scheduling problems to reduce costs related to electricity consumption or pollutants. For instance, Wang et al. (2011) developed a low-carbon production scheduling system by considering using renewable energy. They addressed a single machine scheduling problem by taking into account that carbon is produced due to the power purchased from the grid, as well as the production process, equipment maintenance, and daily activities. Furthermore, Liu (2015) assumed that it is possible to store renewable energy in a battery. He also assumed that carbon is produced only by the power purchased from the grid. He proposed two models for a single machine scheduling problem. Liu (2016) also presented an article with the assumptions of the previous article in conjunction with the goal of minimizing carbon emissions and the earliness and tardiness of jobs. Zhai et al. (2017) carried out research to reduce the cost of power purchased from the grid using wind-generated energy. They proposed a model to reduce the cost of electrical energy for a flow shop scheduling problem by assuming that the electricity generated by the wind energy could be sold to the grid. Next, Biel et al. (2018) developed a bi-objective model in order to minimize the total weighted flow time and the energy cost. Considering the possibility of storing solar energy, Zhang et al. (2017) developed a MIP for a flow shop scheduling problem to reduce the cost of purchasing electricity from the grid. They demonstrated that significant cost savings could be achieved using an optimal scheduling approach. Moreover, Moon and Park (2014), for the first time, examined a flexible job shop scheduling problem that could use both renewable energy and the power grid. They assumed that it was possible to store renewable energy.

2.2. Group scheduling in the flow shop environment

In the group scheduling approach, first proposed by Willey (1975) and extended by Mitrofanov (1966) and Schaller et al. (2000), a group consists of a set of parts that have similar needs for tools, setup, and operation sequence. In cases where the setup of a machine is costly or time-consuming, the idea of grouping parts and doing only one setup for each group can be helpful to increase productivity. In group scheduling, the jobs of one group should be processed sequentially, and at the time of changing jobs from one group to another, a setup time is needed. Based on Bozorgirad and Logendran (2013), group scheduling is usually determined by the scheduling of jobs at two levels. On the one hand, the sequence of groups should be determined according to setup times, and on the other hand, the sequence of jobs should be determined in each group.

Group scheduling in the flow shop environment has many applications in various industries. For instance, in the automobile industry, after painting a group of products, we need to change the color used for the following products. In this case, it is necessary to spend a setup time to clean the environment and prepare the following color (Salmasi et al., 2010). Group scheduling is also used in the blade line in aircraft engine factories, where different types of blades used in aircraft engines are produced. In this production line, the blades that require the same type of processing are placed in a group. If the blades change from one group to another, a setup should be done on the machine, but if the machine moves from one type of blade to another in the same group, a minor adjustment is required whose setup time is negligible. Li (1997) reported that this grouping was implemented many years ago in Pratt and Whitney, a company in the aerospace industry. In addition to these applications, the semiconductor light industry is another group scheduling application in the flow shop environment, described in Section 1.

Regarding the significance of group scheduling in various industries, several papers have addressed this issue. Neufeld et al. (2016) developed a review article for group scheduling in shop environments. They addressed several papers investigating the objective functions such as minimizing the makespan, the total completion time, and the total flow time, but none of these papers has considered the cost of energy consumption. Feng et al. (2018) considered preventive maintenance in a flexible flow shop environment with group scheduling. They aim to minimize the preventive maintenance costs, repair costs, and job tardiness costs. Furthermore, Pan et al. (2020) developed a heuristic algorithm to minimize the makespan in the flow shop scheduling problem with group scheduling assumption. Costa et al. (2020) captured a blocking constraint in the flow shop sequence-dependent group scheduling problem and designed a parallel self-adaptive genetic algorithm to minimize the makespan. Moreover, Cheng et al. (2021) studied a no-wait flow shop group scheduling problem with sequence-dependent setup times. They developed two heuristic algorithms and a local search algorithm to minimize the total completion time.

Table 1 summarizes some of the most related references and indicates the characteristics of each research paper in terms of shop environment, group scheduling, setup time, objective function, and using renewable energy and battery. The symbol "$\surd$" indicates that the corresponding assumption has been considered whereas the symbol "$\times$" implies that assumption has been ignored.

3. Problem description and modeling

3.1. Problem description

In this problem, it is assumed that the parts are produced in the form of group scheduling in a flow shop environment. Each part is called a job, and the processing of each part on a machine is called an operation. A set of similar parts constitute a group, and each machine requires a setup time that depends on the sequence of groups. Therefore, jobs are grouped to increase production efficiency and reduce the number of setup times.

In this study, according to Zhang et al. (2017) and Liu (2015), it is assumed that solar energy can be produced as an on-site renewable energy source using photovoltaic (PV) panels. The electrical energy produced can be used directly to meet electricity needs or stored in a battery for later use. The power consumption of the factory can also be supplied through the grid. It is also possible to store grid power in the battery. Thus, when the cost of electricity is low, electricity can be bought from the grid, and when it is high, the power consumption can be supplied using the battery. In this study, it is assumed that if the battery is fully charged and there is no other demand for power, the electrical energy produced by the PV panels will be wasted. It is also presumed that electrical energy from the PV panels cannot be sold to the grid.

According to Cheng et al. (2017) and Wang et al. (2017), electricity pricing is also considered at different times of the day based on the TOU tariffs. In a TOU tariff, the cost of electricity is determined according to the demand in the electricity grid, in which 24 hours of a day are usually divided into a set of time periods where they are mutually exclusive and jointly exhaustive. These time periods can be categorized as follows: the on-peak hours when the cost of electricity is the highest; the mid-peak hours when the cost of electricity is lower than that of the on-peak hours; and the off-peak hours when the cost of electricity is the lowest. Given the TOU tariff, to avoid high electricity costs, schedules may change in such a way that some jobs might shift from high-priced hours to low-priced ones.

The problem at hand can be denoted using the notations defined by Pinedo (2012) as $F_m|fmls, s^m_{pq}, TOU, re_t, ca|TEC$. In this notation, F_m indicates the flow shop environment with m machines, $fmls$ represents the group scheduling problem; s^m_{pq} indicates the

Table 1
The summary of some of the most related references

Paper	Shop Environment	Group scheduling	Setup time	energy consumption / electricity cost	Renewable Energy	Battery
Mouzon et al. (2010)	single machine	×	×	energy consumption	×	×
Tang et al. (2016)	flexible flow shop	×	×	energy consumption	×	×
Mansouri et al. (2016)	two-machine permutation flow shop	×	√	energy consumption	×	×
Li et al. (2018)	hybrid flow shop	×	√	energy consumption	×	×
Wu and Sun (2018)	flexible job shop	×	×	energy consumption	×	×
Wang et al. (2017)	two-machine permutation flow shop	×	×	electricity cost	×	×
Ho et al. (2020)	two-machine flow shop	×	×	electricity cost	×	×
Schulz et al. (2019)	hybrid flow shop	×	×	electricity cost	×	×
Cui and Lu (2021)	flow shop	×	×	electricity cost	×	×
Wang et al. (2011)	single machine	×	×	energy consumption	√	√
Liu (2015)	single machine	×	×	energy consumption	√	√
Liu (2016)	single machine	×	×	energy consumption	√	√
Zhai et al. (2017)	flow shop	×	×	electricity cost	√	×
Biel et al. (2018)	flow shop	×	×	electricity cost	√	×
Zhang et al. (2017)	hybrid flow shop	×	×	electricity cost	√	√
Moon and Park (2014)	flexible job shop	×	×	electricity cost	√	√
Schaller et al. (2000)	flowline	√	√	×	×	×
Salmasi et al. (2010)	flow shop	√	√	×	×	×
Bozorgirad and Logendran (2013)	hybrid flow shop	√	√	×	×	×
Feng et al. (2018)	flexible flow shop	√	√	×	×	×
Pan et al. (2020)	flow shop	√	√	×	×	×
Costa et al. (2020)	flow shop	√	√	×	×	×
Cheng et al. (2021)	no-wait flow shop	√	√	×	×	×

sequence-dependent setup time between groups p and q on machine m; TOU alludes to electricity tariff in the schedule; re_t indicates the renewable energy; ca represents the capacity of the battery that stores renewable energy, and finally TEC represents the objective function, which implies the total energy cost (TEC). Other assumptions are as follows.

- All machines and jobs are available at the beginning of the scheduling horizon.
- All machines have two modes: busy mode and idle (shut down) mode, where in the latter, no energy is consumed.
- All operations of each group should be completed before the end of the time horizon.
- Preemption is not allowed during processing any job.
- The jobs of one group should be processed one after the other without interruption by other groups.
- The buffer capacities between machines are unlimited.
- Setting up of each machine for processing a new group of jobs can be done before jobs of that group are available on the machine. Furthermore, the setup time of a job (if needed) and its processing time are non-interrupted.
- Setup time is only considered for groups. This implies that there is no setup time between two consecutive jobs of the same group.
- No power is consumed during setup times.
- The sequence of groups and all jobs belonging to a group cannot be changed on different machines implying the permutation flow shop scheduling.
- The power required by each machine can be supplied through two sources of energy: solar renewable energy and power grid.
- The battery has a constant capacity and can be charged using both solar panels and power at the grid.
- The battery, as a source of energy, is empty at the beginning of the scheduling horizon.

The goal of solving $F_m|fmls, s^m_{pq}, TOU, re_t, ca|TEC$ is to minimize the cost of purchasing electricity from the grid. To optimize it, decisions should be made about the sequence and scheduling of groups and jobs in each group, as well as whether or not each machine is idle in each time period. In the following, we develop two mixed-integer linear programming models and a decomposition-based heuristic algorithm to solve the problem.

3.2. A time-unit index model

In this section, we develop a time-unit index model, dubbed TUIM, for $F_m|fmls, s^m_{pq}, TOU, re_t, ca|TEC$. Since electricity costs depend on consumption time, the time-unit index (one minute) is used in the TUIM to define the variables of the model. Zhang et al. (2014), Wang et al. (2017), and Schulz et al. (2019) considered electricity tariffs and adopted a similar approach to model the flow shop. The parameters and variables of the TUIM are defined in Table 2.

The TUIM reads as follows.

$$TUIM : Min\ TEC = \sum_{t\in T} r_t.EG_t \tag{1}$$

Subject to:

$$\sum_{q\in G\backslash\{p,0\}} Y_{pq} = 1,\ \ \forall p \in G\backslash\{g+1\}; \tag{2}$$

$$\sum_{p\in G\backslash\{q,g+1\}} Y_{pq} = 1, \forall q \in G\backslash\{0\}; \tag{3}$$

$$Y_{pq} + Y_{qp} \le 1,\ \ \forall\ p, q \in G : p < q; \tag{4}$$

$$\left(1 - U_{plj}\right)\beta + \sum_{t\in T} t\ Z^m_{pjt} - \sum_{t\in T} t\ Z^m_{plt} \ge \pi^m_{pl}, \quad \forall m \in M; \forall p \in G; \forall l, j \in J_p : l < j; \tag{5}$$

$$\beta U_{plj} + \sum_{t\in T} t\ Z^m_{plt} - \sum_{t\in T} t\ Z^m_{pjt} \ge \pi^m_{pj},\ \ \forall m \in M; \quad \forall p \in G; \forall l, j \in J_p : l < j; \tag{6}$$

Table 2
The parameters and variables of the TUIM

Sets	Definition
$G = \{0, 1, 2, \ldots g, g+1\}$	The set of groups, indexed by p and q. Groups 0 and *g+1* are dummy and each contains only one dummy job with zero processing time and zero power consumption.
$M = \{1, 2, \ldots, \mu\}$	The set of machines, indexed by m.
$J_p = \{1, 2, \ldots n_p\}$	The set of jobs of group p, indexed by l and j.
$T = \{1, 2, \ldots T_{max}\}$	The set of time units (minutes), indexed by t and t'.
Parameters	**Definition**
g	The number of groups.
n_p	The number of jobs in group p.
μ	The number of machines.
s^m_{pq}	Setup time between groups p and q on machine m.
π^m_{pj}	Processing time of job j from group p on machine m.
pe^m_{pj}	Power consumption of job j from group p when it is processed on machine m
r_t	Electricity price at time unit t.
ca	Battery capacity.
re_t	The electrical energy produced by the renewable energy source at time unit t.
T_{max}	Scheduling horizon (minutes).
β	A very large number.
Variables	**Definition**
X^m_{pjt}	It takes the value of one if job j of group p is processed on machine m at time unit t, otherwise, it takes zero.
Z^m_{pjt}	It takes the value of one if the processing of job j from group p on machine m starts at the beginning of time unit t, otherwise, it takes zero.
Y_{pq}	It takes the value of one if group q is processed immediately after group p, otherwise, it takes zero.
U_{plj}	It takes the value of one if in group p, job j is processed after job l (consecutively or disjointedly), otherwise, it takes zero.
CG^m_p	Completion time of processing of all jobs in group p on machine m.
EB_t	The energy stored in the battery at the beginning of time unit t (assume that $EB_1 = 0$)
EG_t	Electricity purchased from the grid at unit time t.
TEC	Total electricity cost purchased from the grid.

$$(1 - Y_{pq})\beta + \sum_{t\in T}(t-1)Z^m_{qlt} - CG^m_p \geq s^m_{pq},\ \ \forall m \in M;\ \forall p \in G\backslash\{g+1\}; \forall q \in G\backslash\{0, p\}; \forall l \in J_q; \tag{7}$$

$$CG^m_p \geq \sum_{t\in T}(t-1)\ Z^m_{pjt} + \pi^m_{pj},\ \ \forall m \in M; \forall p \in G; \forall j \in J_p; \tag{8}$$

$$\sum_{t\in T} t\ Z^m_{pjt} - \sum_{t\in T} t\ Z^{m-1}_{pjt} \geq \pi^{m-1}_{pj},\ \ \forall p \in G; \forall j \in J_p; \forall m \in M\backslash\{1\}; \tag{9}$$

$$CG^{\mu}_p \leq T_{max},\ \ \forall p \in G; \tag{10}$$

$$\sum_{t\in T} Z^m_{pjt} = 1, \forall m \in M; \forall p \in G; \forall j \in J_p; \tag{11}$$

$$\sum_{p\in G}\sum_{j\in J_p} X^m_{pjt} \leq 1, \forall m \in M; \forall t \in T; \tag{12}$$

$$\sum_{t'=t}^{min\left(t+\pi^m_{pj}-1, T_{max}\right)} X^m_{pjt'} \geq Z^m_{pjt}\ \pi^m_{pj}, \forall m \in M; \forall p \in G; \forall j \in J_p; \forall t \in T; \tag{13}$$

$$\sum_{t\in T} X^m_{pjt} = \pi^m_{pj},\ \ \forall m \in M; \forall p \in G; \forall j \in J_p; \tag{14}$$

$$EB_t \leq ca,\ \ \forall t \in T; \tag{15}$$

$$EG_t \geq \sum_{m\in M}\sum_{p\in G}\sum_{j\in J_p} pe^m_{pj}\ X^m_{pjt} - EB_t - re_t,\ \ t = T\,max; \tag{16}$$

$$EB_t \leq EB_{t-1} + re_{t-1} + EG_{t-1} - \sum_{m\in M}\sum_{p\in G}\sum_{j\in J_p} pe^m_{pj}\ X^m_{pj(t-1)}, \forall t \in T\backslash\{1\}; \tag{17}$$

$$X^m_{pjt} \in \{0, 1\}, \forall m \in M; \forall p \in G; \forall j \in J_p; \forall t \in T \tag{18}$$

$$Y_{pq} \in \{0, 1\},\ \ \forall p, q \in G \tag{19}$$

$$Z^m_{pjt} \in \{0, 1\},\ \ \forall m \in M; \forall p \in G; \forall j \in J_p; \forall t \in T \tag{20}$$

$$U_{plj} \in \{0, 1\},\ \ \forall p \in G; \forall\ l, j \in J_p \tag{21}$$

$$CG^m_p \geq 0,\ \ \forall m \in M; \forall p \in G \tag{22}$$

$$EG_t \geq 0,\ \ \forall t \in T \tag{23}$$

$$EB_t \geq 0,\ \ \forall t \in T \tag{24}$$

Objective function (1) represents minimizing the cost of purchasing electricity from the grid. Constraints (2) to (4) indicate the sequence of groups on each machine. These constraints state that only one group can be processed immediately after and before each group. Constraints (5) and (6) limit the start time of jobs in each group in such a way that the start time of each job must be equal to or larger than the completion time of the previous job. Constraints (7) state that the start time of each job on each machine is equal to or larger than the completion time of the previous group plus the corresponding setup time. Constraints (8) show that the completion time of a group must be equal to or larger than the total completion time of jobs belonging to that group. Constraints

(9) indicate that each job can be processed on a machine when its process is finished on the previous machine. According to Constraints (10), the completion time of all groups on the last machine must be equal to or less than the scheduling horizon. Due to Constraints (11), each job has only one start time on each machine, i.e., each job is processed only once. Constraints (12) impose that in each time unit, at most, one job can be processed on each machine. Constraints (13) state that whenever a job is started, it must be continued until it is completed, i.e., preemption is not allowed. Given that the objective function is to minimize electricity cost, this constraint always seems to have a lower bound, but it is not so. For example, in the time units when the battery capacity is full, and there is a surplus of solar energy, X^m_{pjt} may tend to take a value more than the lower bound without affecting the cost of electricity purchased from the grid. Therefore, this problem is solved by considering the constraints (14). This set of constraints ensures that the number of time units at which a job is being performed has to be equal to the processing time of that job on the entire scheduling horizon. Constraints (15) to (17) indicate the amount of electricity purchased from the grid and stored in the battery. Constraints (15) impose that the electrical energy stored in the battery must not exceed the rated capacity of the battery. Constraints (16) indicate the minimum amount of electricity purchased from the grid in the last time unit. Constraints (17) calculate the electricity stored in the battery for each time unit, according to the electricity purchased from the grid in the previous time unit and the amount of solar power in the previous time unit. Due to this limitation, it is possible to purchase more electricity from the grid and store the excess purchased electricity in the battery and consume it when the cost of purchasing electricity from the grid is higher. Finally, Constraints (18) to (24) indicate the type of variables used in this model.

3.3. A time-interval index model

In this section, we develop a time-interval index model, dubbed TIIM, for $F_m|fmls, s^m_{pq}, TOU, re_t, ca|TEC$. In the TUIM, the time-unit index, which is equal to one minute, is used to define the variables, so this model is strongly dependent on the duration of the scheduling horizon. As the scheduling horizon increases, the number of variables and the complexity of the problem surge. To reduce the dependence of the number of variables on the scheduling horizon, we develop the TIIM in which the formulation idea is based on time intervals. In this new modeling, the scheduling horizon is divided into a set of mutually exclusive, jointly exhaustive and equal-length intervals. These intervals should be such that the cost of electricity purchased from the grid does not change during an interval. It should be noted that changes are allowed in electricity prices between intervals.

The TIIM consists of two parts. In the first part, based on a position assignment approach, the sequence and schedule of groups and jobs are determined. We assume that there are g positions, and each group is assigned to one of them. Therefore, the group in the first position is processed first, and then the group in the second position is processed, and so on until the last group. Furthermore, the sequence of jobs of each group and the start and completion time of each group and each job are fixed in this part of the model. In the second part of the model, the scheduling horizon is divided into a number of time intervals. In this part, according to the start and completion time of each job, obtained from the first part, a particular time interval is determined for processing of each job.

The critical point is how to produce renewable energy in each interval. Renewable energy is being produced continuously and is accumulated over each time period. In other words, the whole renewable energy produced within an interval is not available at the beginning of that time interval, and we cannot consume all of it. Therefore, for simplification, we assume that the renewable energy produced in each time interval cannot be used in the same period, and this energy should first be stored in the battery and then used in the subsequent periods. It should be noted that this assumption only applies to the TIIM in which the time index is longer than one minute. In the TUIM, because time is in minutes, it can be roughly assumed that renewable energy produced in one minute is available from the beginning. For example, in the TIIM, if the length of the intervals is 60 minutes and it is possible to use renewable energy in the same interval, the whole energy produced in these 60 minutes may be consumed in the first few minutes of this interval, whereas it might not be available at those times.

To better understand the TIIM, suppose that we want to schedule two groups, each consisting of three jobs. Consider a 180-minute scheduling horizon, divided into three time intervals of 60 minutes. Fig. 2 shows the schedule of the jobs for these two groups, which is obtained by the first part of the TIIM. According to this figure, all jobs related to the group in the first position are processed in the second interval, and we consider the price of electricity corresponding to the second interval for them. The renewable energy that we can use to process these jobs is equal to the amount produced before the start of the second interval and is available in the battery. As shown in Fig. 2, the jobs for the group in the second position are processed in both time intervals 2 and 3. Job 3 of group 2 starts at time $\tau = 110$ with the processing time of 10 minutes and finishes at time $\tau = 120$, i.e., this job was processed in the second interval. Thus, we consider the price of the second interval to process this job, but jobs 1 and 2 are processed in the third interval. Hence, we consider the price of electricity in the third interval to process these jobs. In addition, job 3 of group 2 is allowed to use only the renewable energy stored in the battery until the beginning of the second time interval, whereas jobs 1 and 2 of the same group are allowed to use the renewable energy stored in the battery until the beginning of the third time interval.

It should be noted that the globally optimal solution of the TIIM is obtained in the case that the length of each interval is one minute. For cases where the interval length is more than one minute, the optimal solution obtained from the model may not be the global optimal, but only a locally optimal solution. In other words, as the length of the intervals increases, the number of variables and the complexity of the model decrease, but instead, its accuracy decreases slightly. Moreover, we consider some dummy jobs in some groups to have an equal number of jobs in all groups. This assumption decreases the number of variables in the TIIM because we do not have to consider the group index for assigning jobs to positions. In Table 3, the parameters and variables are defined for the TIIM.

The TIIM reads as follows.

$$TIIM: Min\ TEC = \sum_{\tau \in \mathrm{T}} r_\tau . EG_\tau \tag{25}$$

Subject to:

$$\sum_{i \in I} W_{ip} = 1,\ \ \forall p \in G^*; \tag{26}$$

$$\sum_{p \in G^*} W_{ip} = 1,\ \ \forall i \in I; \tag{27}$$

$$W_{00} = 1 \tag{28}$$

$$\sum_{p \in G^*} \sum_{q \in G^* \backslash \{p,0\}} \mathrm{Y}_{ipq} = 1,\ \ \forall i \in I \backslash \{g\}; \tag{29}$$

$$\mathrm{Y}_{ipq} \le W_{ip},\ \ \forall i \in I \backslash \{g\}; \forall p \in G^*; \forall q \in G^* \backslash \{p, 0\}; \tag{30}$$

$$\mathrm{Y}_{ipq} \le W_{i+1\ q},\ \ \forall i \in I \backslash \{g\}; \forall p \in G^*; \forall q \in G^* \backslash \{p, 0\}; \tag{31}$$

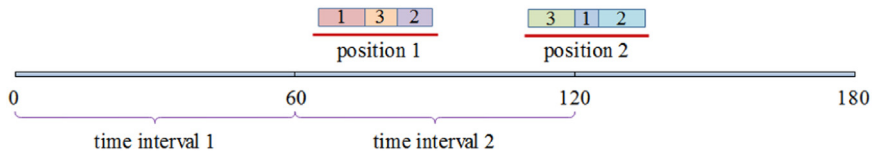

Fig. 2. Illustration of the TIIM modeling approach.

Table 3
The parameters and variables of the TIIM

Sets	Definition
$G^* = \{0, 1, 2, \ldots g\}$	The set of groups, indexed by p and q. Group 0 is dummy and contains only one job.
$I = \{0, 1, 2, \ldots g\}$	The set of positions of groups, indexed by i (position 0 is dummy).
$N = \{1, 2, \ldots b_{max}\}$	The set of jobs in each group, indexed by l and j. The number of jobs in group p is equal to n_p and $max(0, b_{max} - n_p)$ is equal to the number of dummy jobs in group p.
$\mathrm{T} = \{1, 2, \ldots, k\}$	The set of time intervals, indexed by τ and τ'.
Parameters	**Definition**
$b_{max} = \max\{n_1, \ldots, n_p\}$	Maximum number of jobs in each group
k	The number of time intervals.
d	The length of each time interval (minutes).
Variables	**Definition**
F_{ij}^m	Completion time of job j when its group is placed in position i on machine m.
W_{ip}	If group p is placed in position i, it takes the value of one, otherwise zero.
Q_{ijl}	If job l is processed after job j (immediately or disjointedly) while both belong to the same group and that group is placed in position i, it takes the value of one, otherwise zero.
Y_{ipq}	If group p is placed in position i and is processed immediately before group q, it takes the value of one, otherwise zero.
C_i^m	Completion time of all jobs processed in position i on machine m.
$X_{pj\tau}^m$	If the whole or a part of job j of group p is processed on machine m in interval τ, it takes the value of one, otherwise zero.
$V_{pj\tau}^m$	The amount of time that job j of group p is processed on machine m in interval τ.

$$F_{ij}^m - F_{il}^m \geq -\beta \mathrm{Q}_{ijl} + \sum_{p \in G^* \{0\}} \pi_{pj}^m W_{ip}, \ \forall m \in M; \quad \forall i \in I \backslash \{0\}; \ \forall j, l \in N : j < l; \tag{32}$$

$$F_{il}^m - F_{ij}^m \geq \beta\left(\mathrm{Q}_{ijl} - 1\right) + \sum_{p \in G^* \backslash \{0\}} \pi_{pl}^m W_{ip}, \quad \forall m \in M; \forall i \in I \backslash \{0\}; \forall \ j, l \in N : j < l; \tag{33}$$

$$C_0^m = 0, \forall m \in M; \tag{34}$$

$$F_{ij}^m \geq C_{i-1}^m + \sum_{p \in G^*} \sum_{q \in G^* \backslash \{0, p\}} \mathrm{Y}_{(i-1)pq} s_{pq}^m + \sum_{p \in G^* \backslash \{0\}} \pi_{pj}^m W_{ip}, \quad \forall m \in M; \forall i \in I \backslash \{0\}; \forall j \in N; \tag{35}$$

$$F_{ij}^m \geq F_{ij}^{m-1} + \sum_{p \in G^* \backslash \{0\}} \pi_{pj}^m W_{ip}, \ \forall m \in M \backslash \{1\}; \forall i \in I \backslash \{0\}; \forall j \in N; \tag{36}$$

$$C_i^m \geq F_{ij}^m, \ \forall m \in M; \forall i \in I \backslash \{0\}; \forall j \in N; \tag{37}$$

$$C_g^\mu \leq kd \tag{38}$$

$$\sum_{\tau \in \mathrm{T}} V_{pj\tau}^m = \pi_{pj}^m, \ \forall m \in M; \forall p \in G^* \backslash \{0\}; \forall j \in N; \tag{39}$$

$$V_{pj\tau}^m \leq \pi_{pj}^m X_{pj\tau}^m, \ \forall \tau \in \mathrm{T}; \forall m \in M; \forall p \in G^* \backslash \{0\}; \forall j \in N; \tag{40}$$

$$F_{ij}^m \geq ((\tau - 1) * d + 1) - \beta\left(2 - X_{pj\tau}^m - W_{ip}\right), \ \forall \tau \in \mathrm{T} \backslash \{1\}; \quad \forall m \in M; \forall p \in G^* \backslash \{0\}; \forall j \in N; \forall i \in I \backslash \{0\}; \tag{41}$$

$$F_{ij}^m - \pi_{pj}^m \leq (\tau d - 1) + \beta\left(2 - X_{pj\tau}^m - W_{ip}\right), \ \forall \tau \in \mathrm{T}; \quad \forall m \in M; \forall p \in G^* \backslash \{0\}; \forall j \in N; \forall i \in I \backslash \{0\}; \tag{42}$$

$$V_{pj\tau}^m \leq F_{ij}^m - (\tau - 1)d + \beta\left(2 - X_{pj\tau}^m - W_{ip}\right), \forall \tau \in \mathrm{T} \backslash \{1\}; \quad \forall m \in M; \forall p \in G^* \backslash \{0\}; \forall j \in N; \forall i \in I \backslash \{0\}; \tag{43}$$

$$V_{pj\tau}^m \leq \tau d - \left(F_{ij}^m - \pi_{pj}^m\right) + \beta\left(2 - X_{pj\tau}^m - W_{ip}\right), \forall \tau \in \mathrm{T}; \quad \forall m \in M; \forall p \in G^* \backslash \{0\}; \forall j \in N; \forall i \in I \backslash \{0\}; \tag{44}$$

$$EB_\tau \leq ca, \forall \tau \in \mathrm{T}; \tag{45}$$

$$EG_\tau \geq \sum_{m \in M} \sum_{p \in G^* \backslash \{0\}} \sum_{j \in N} pe_{pj}^m V_{pj\tau}^m - EB_\tau, \ \forall \tau \in \mathrm{T}; \tag{46}$$

$$EB_{\tau+1} \leq EB_\tau + EG_\tau + re_\tau - \sum_{m \in M} \sum_{p \in G^* \backslash \{0\}} \sum_{j \in N} pe_{pj}^m . V_{pj\tau}^m, \ \forall \tau \in \mathrm{T} \backslash \{k\}; \tag{47}$$

$$W_{ip} \in \{0, 1\}, \ \forall p \in G^*; \forall i \in I; \tag{48}$$

$$\mathrm{Q}_{ijl} \in \{0, 1\}, \ \forall i \in I; \forall j, l \in N; \tag{49}$$

$$\mathrm{Y}_{ipq} \in \{0, 1\}, \ \forall i \in I; \forall p, q \in G^*; \tag{50}$$

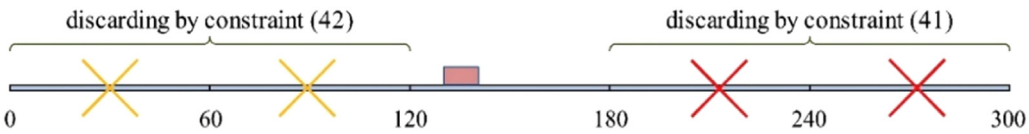

Fig. 3. Explanation of Constraints (41) and (42).

$$X_{pj\tau}^{m} \in \{0,1\},\ \forall m \in M; \forall p \in G^{*}; \forall j \in N; \forall \tau \in \mathrm{T}; \tag{51}$$

$$F_{ij}^{m} \geq 0,\ \forall m \in M; \forall i \in I; \forall j \in N; \tag{52}$$

$$C_{i}^{m} \geq 0,\ \forall m \in M; \forall i \in I; \tag{53}$$

$$V_{pj\tau}^{m} \geq 0,\ \forall m \in M; \forall p \in G^{*}; \forall j \in N; \forall \tau \in \mathrm{T}; \tag{54}$$

$$EG_{\tau} \geq 0,\ \forall \tau \in \mathrm{T}; \tag{55}$$

$$EB_{\tau} \geq 0,\ \forall \tau \in \mathrm{T}; \tag{56}$$

Objective function (25) represents minimizing the cost of purchasing electricity from the grid. Constraints (26) to (38) are related to the first part of the model, and Constraints (39) to (48) pertain to the second part of the model. In the first part, Constraints (26) and (27) show that each group can only be placed in one position and vice versa. According to Constraints (28), the zero dummy group is placed in the zero dummy position. Constraints (29) to (31) determine the sequence of the groups. Constraints (29) state that only one group is placed in each position i, and only one other group is placed immediately after that. According to Constraints (30) and (31), the variable Y_{ipq} will equal one if group p is assigned to position i and group q to position $i+1$. Constraints (32) and (33) link the completion times of jobs to their sequences. According to these constraints, the completion time of any job in a group belonging to a specific position must be equal to or larger than the completion time of jobs previously performed. Constraints (34) show that the completion time of the dummy position is zero on all machines. Constraints (35) indicate that a job of a particular group in a position is permitted to be processed when the processing of the group placed in the previous position is completed, and the setup time between the job belonging to the current group and the previous one has also passed. According to Constraints (36), any job can be processed on a new machine when it has finished the required processing on the previous machine. Constraints (37) state that the completion time of each group must be equal to or longer than the completion time of all jobs belonging to that group. According to Constraints (38), the completion time of the group in the last position on the last machine must be equal to or less than the scheduling horizon where kd in this model corresponds to T_{max}in the previous model.

In the second part of the model, Constraints (39) state that the total time spent on processing each job in all time intervals must be equal to the processing time of that job. According to Constraints (40), if a job is assigned to an interval, the whole or a part of that job can be processed in that interval. Constraints (41) and (42) determine interval(s) in which each job is processed based on the start and completion time of the job. For example, consider Fig. 3 in which a job, shown as a small red box, is going to be scheduled. Assume that the completion time of the job is at $t = 140$, and the processing time of this job is equal to 10 minutes. Due to Constraints (41), this job is allowed to be processed in interval 3 or before, but it is not allowed to be processed in the intervals after interval 3. In addition, due to the Constraints (42) and as the job starts at $t = 130$, this job is permitted to be processed in interval 3 and subsequent ones, but the intervals before interval 3. Therefore, considering both constraints simultaneously, this job can only be processed in interval 3.

Constraints (43) and (44) determine the amount of processing time of each job in each time interval. Fig. 4 shows two different modes of placing a job at time intervals. In case 1, according to Constraints (41) and (42), the desired job is processed only in interval 4. In this figure, the results of the expressions on the right side of Constraints (43) and (44) are denoted by ϑ_1 and ϑ_2, respectively. According to Constraints (39), (43), and (44), the processing time of this job in interval 4 is equal to the processing time of that job. In case 2, according to Constraints (41) and (42), the job is permitted to be processed in two intervals 4 and 5. In this case, the outcome of the expression on the right side of Constraints (43) is indicated by ϑ_1' and ϑ_1'' and the result of the expression on the right side of Constraint (44) is denoted by ϑ_2' and ϑ_2''. Consequently, on the basis of Constraints (39), (43) and (44), the processing time of the desired job will be ϑ_2' in interval 4 and ϑ_2' in interval 5.

Constraints (45) to (47) show the amount of electricity purchased from the grid and the amount of solar energy stored in the battery. Constraints (45) limit the electrical energy stored in the battery to its rated capacity. Constraints (46) impose the minimum amount of electricity purchased from the grid at any time. They prevent using the renewable energy produced in an interval in the same interval. Constraints (47) calculate the amount of electricity stored in the battery for the next interval based upon the amount of electricity purchased from the grid and the amount of solar power available in the current interval. Due to Constraints (46), it is possible to buy more electricity from the grid, and based on Constraints (47), the excess electricity purchased from the grid is stored in the battery to be consumed later. Finally, Constraints (48) to (56) indicate the type of variables used in this model.

If we consider $d = 1$, then the TIIM can be more simplified by removing the variables $V_{pj\tau}^{m}$ and constraints (40), (43), (44), and (54). In this scenario, Constraints (39) is replaced by Constraints (57).

$$\sum_{\tau \in \mathrm{T}} X_{pj\tau}^{m} = \pi_{pj}^{m},\ \forall m \in M; \forall p \in G^{*} \backslash \{0\}; \forall j \in N; \tag{57}$$

In addition, in this scenario, the TIIM is similar to the TUIM; therefore, to obtain the amount of electricity purchased from the grid, it can be assumed that it is possible to use renewable energy in the same time interval when this energy is produced. Thus, Constraints (46) and (47) are replaced by Constraints (16) and (17), except that $j \in N$, $p \in G^{*} \backslash \{0\}$ and $\tau \in \mathrm{T}$.

4. A decomposition-based heuristic algorithm

Fang et al. (2016) proved that the single-machine scheduling problem considering the TOU tariff and constant speed of the machine is strongly NP-hard. Since $F_m | fmls, s_{lpi}^{k}, TOU, re_t, ca | TEC$ is a generalization of the mentioned problem, it has at least the same degree of complexity. Therefore, to reduce the CPU runtime, in this

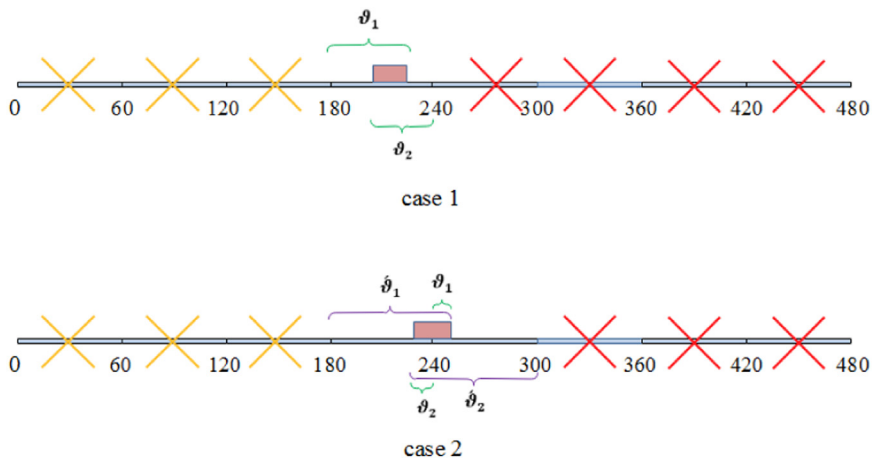

Fig. 4. Explanation of Constraints (43) and (44).

Table 4
The parameters and variables of the DBHA

Sets	Definition
$\Phi = (\sigma_0, \sigma_1, \ldots, \sigma_g)$	A sequence of all groups (dummy group 0 is placed at the beginning of this list and contains only a dummy job).
$\Phi^{(p)} = (\sigma_0, \sigma_1, \ldots, \sigma_p)$	The sequence of groups σ_0 to σ_p, created based on sequence Φ.
$\Phi^{\{p\}} = \{\sigma_0, \sigma_1, \ldots, \sigma_p\}$	The set of groups in sequence $\Phi^{(p)}$.
$\bar{\Phi}^{(p)} = (\sigma_{p+1}, \ldots, \sigma_g)$	The sequence of groups σ_{p+1} to σ_g, created based on sequence Φ.
$\bar{\Phi}^{\{p\}} = \{\sigma_{p+1}, \ldots, \sigma_g\}$	The set of groups in sequence $\bar{\Phi}^{(p)}$.
$\Gamma_p = (\delta_1^p, \ldots, \delta_{n_p}^p)$	A sequence of jobs in group p.
$\Gamma_p^{(j)} = (\delta_1^p, \ldots, \delta_j^p)$	The sequence of jobs δ_1^p to δ_j^p in group p, created based on sequence Γ_p.
$\Gamma_p^{\{j\}} = \{\delta_1^p, \ldots, \delta_j^p\}$	The set of jobs in sequence $\Gamma_p^{(j)}$.
Parameters	**Definition**
λ	Index of groups.
α	Index of jobs.
σ_i	The group that is placed in position i_{th} based on sequence Φ.
ς_p	The position of group p based on sequence Φ.
δ_i^p	A job from group p that is placed in position i^{th} based on sequence Γ_p.
ξ_j^p	The position of job j from group p based on sequence Γ_p.
π_p^m	The processing time of group p on machine m.
$\pi_i''^m$	The processing time of a group or job that is placed in position i on machine m.
$p_p'^m$	The power consumption of group p that is being processed on machine m.
$pe_i''^m$	The power consumption of the group or job placed in the position i and being processed on machine m.
e_p'	The weighted average power consumption of group p on all machines.
e_j''	The weighted average power consumption of job j on all machines.
Variables	**Definition**
$F_p'^m$	The completion time of group p on machine m (used in model M_1).
$F_i''^m$	The completion time of the job placed in position i on machine m (used in model M_2).
$F_{pj}'''^m$	The completion time of job j in group p on machine m (used in model M_3).
Q_{pq}'	If group q is processed (immediately or disjointedly) after group p, it takes the value of one, otherwise zero (used in model M_1).
Q_{ih}''	If a job or group is processed in position h (immediately or disjointedly) after the job or group placed in position i, it takes the value of one, otherwise zero (used in model M_2).
$X_{p\tau}'^m$	If the whole or a part of group p is processed on machine m during interval τ, it takes the value of one, otherwise zero (used in model M_1).
$X_{i\tau}''^m$	If the whole or a part of the job or group in the position i is processed on machine m during interval τ, it takes the value of one, otherwise zero (used in model M_2).
$V_{p\tau}'^m$	The amount of time that group p is processed on machine m during interval τ (used in model M_1).
$V_{i\tau}''^m$	The amount of time that the job or group placed in position i is processed on machine m during interval τ (used in model M_2).

section, we develop a DBHA. The structure of this algorithm is based on the well-known NEH method developed by Nawaz et al. (1983). The parameters and variables used in the DBHA are listed in Table 4.

4.1. The general sketch of the DBHA

The general sketch of the DBHA is presented in Algorithm 1. The DBHA decomposes the main problem into three phases, dubbed model M_1, model M_2, and model M_3. In the first phase, each group is considered as a single job, and its processing time on each machine equals the total processing time of all jobs included in the group plus the average setup time for starting that group. In addition, the power consumption for each group is calculated as the weighted average power consumption of the jobs involved in the group. In this phase, first, the groups are arranged in descending order based on their power consumption, then the relative position of the two groups with the most power consumption is determined by model M_1. The relative position of each new group in the sequence is determined by re-solving model M_1 in such a way

Algorithm 1 The pseudocode of the DBHA.

1. For $\forall p \in G^*\backslash\{0\}, m \in M$, let $\acute{\pi}_p^m = \sum_{j\epsilon J_p} \pi_{pj}^m + \frac{\sum_{q\epsilon G^*\backslash\{p\}} s_{qp}^m + \sum_{q\epsilon G^*\backslash\{0,p\}} s_{pq}^m}{2*g-1}$ and $\acute{pe}_p^m = \sum_{j\epsilon J_p} PE_{pj}^m \pi_{pj}^m / \acute{\pi}_p^m$.
2. For $\forall p \in G^*\backslash\{0\}$, let $\acute{e}_p = \sum_{m\epsilon M} \acute{pe}_p^m \acute{\pi}_p^m / \sum_{m\epsilon M} \acute{\pi}_p^m$. Arrange the groups based on descending order of $\acute{e}_p$ as $\Phi = (\sigma_1, \sigma_2, \dots, \sigma_g)$. Next, put σ_0 in sequence $\Phi = (\sigma_0, \sigma_1, \dots, \sigma_g)$ where $\acute{\pi}_0^m = 0$ and $\acute{pe}_0^m = 0$.

For $\lambda = 2$ to g do

3. Apply model M_1 to the set of groups of $\Phi^{\{\lambda\}}$ and based on sequence $\Phi^{(\lambda-1)}$ to update sequence $\Phi = (\sigma_0, \sigma_1, \sigma_2, \dots, \sigma_g)$.

End

4. For $\forall p \in \Phi^{\{g\}}\backslash\{0\}, m \in M$, if $\varsigma_{p-1} = \varsigma_p - 1$, update values of $\acute{\pi}_p^m$ and $\acute{pe}_p^m$ as $\acute{\pi}_p^m = \sum_{j\epsilon J_p} \pi_{pj}^m + s_{(p-1)p}^m$ and $\acute{pe}_p^m = \sum_{j\epsilon J_p} pe_{pj}^m \pi_{pj}^m / \acute{\pi}_p^m$.

For $\lambda = 1$ to g do

5. For $\forall j \in J_{\sigma_\lambda}$, let $\acute{e}_j = \sum_{m\epsilon M} pe_{\sigma_\lambda j}^m \pi_{\sigma_\lambda j}^m / \sum_{m\epsilon M} \pi_{\sigma_\lambda j}^m$. Arrange the jobs of group σ_λ based on descending order of $\acute{e}_j$ as $\Gamma_{\sigma_\lambda} = (\delta_1^{\sigma_\lambda}, \dots, \delta_{n_{\sigma_\lambda}}^{\sigma_\lambda})$.

For $\alpha = 2$ to n_{σ_λ} do

6. **For** $\forall p \in \Phi^{\{\lambda-1\}}, m \in M$, let $\acute{\pi}_{\varsigma_p}^m = \acute{\pi}_p^m$ and $\acute{pe}_{\varsigma_p}^m = \acute{pe}_p^m$.
7. **For** $\forall j \in \Gamma_{\sigma_\lambda}^{\{\alpha\}}, m \in M$, determine position i in such a way that $i = \lambda - 1 + \xi_j^{\sigma_\lambda}$ is held and let $\acute{\pi}_i^m = \pi_{\sigma_\lambda j}^m$ and $\acute{pe}_i^m = pe_{\sigma_\lambda j}^m$.
8. **For** $\forall p \in \overline{\Phi}^{\{\lambda\}}, m \in M$, determine position i in such a way that $i = \varsigma_p + \alpha - 1$ is held and let $\acute{\pi}_i^m = \acute{\pi}_p^m$ and $\acute{pe}_i^m = \acute{pe}_p^m$.
9. Apply model M_2 and update sequence $\Gamma_{\sigma_\lambda} = (\delta_1^{\sigma_\lambda}, \dots, \delta_{n_{\sigma_\lambda}}^{\sigma_\lambda})$.

End

End

10. Apply model M_3 to obtain the final schedule.

that the relative order of previously determined groups remains unchanged. This process continues until the sequence of all groups is determined.

In step 1 of Algorithm 1, each group is considered as a job with a specific processing time and power consumption. The processing time of each group is calculated as the total processing time of jobs involved in that group, and the power consumption of the group is calculated as the weighted average of the power consumption of jobs belonging to the group. In this step, the sequence and setup time of each group is not exactly known, so the setup time is considered as the average of the setup times for each group, and this value is added to the processing time of that group. In step 2, the groups are temporarily arranged based on the descending order of their power consumption obtained from the weighted average power consumption of each group on all machines. It should be noted that this sequence is not the final sequence of groups, and we benefit from it to choose groups one by one and determine their final relative order using model M_1. In step 3, based on each run of model M_1, the relative position of a group is determined among the groups whose relative position has already been fixed. Since the relative order of groups that have already been determined should not be changed, each time model M_1 is run, the relative position of the previously fixed groups is given as a constant input to model M_1. In the last run of model M_1, a complete sequence of all groups is obtained. In step 4, the processing time and power consumption of each group are calculated and updated based on the obtained final sequence.

In step 5, the jobs of each group are temporarily arranged based on the power consumption obtained from the weighted average power consumption of each job on all machines. In steps 6 to 8, the groups are divided into three categories. The first category (step 6) involves the groups whose job sequence is determined (set $\Phi^{\{\lambda-1\}}$). In this category, each group is considered as a job, and their processing time and power consumption are determined in step 4. The positions of these jobs are fixed using model M_2 and should be scheduled before other jobs in the same way as determined in sequence $\Phi^{(\lambda-1)}$. The second category (step 7) includes the jobs belonging to the group whose sequence of jobs is being determined (group σ_λ). The relative position of jobs $\Gamma_{\sigma_\lambda}^{\{\alpha-1\}}$ is fixed in model M_2, and based on determined sequence $\Gamma_{\sigma_\lambda}^{(\alpha-1)}$, they are placed after the jobs of the first category. The relative position of job α is unknown in this sequence, determined through model M_2. The third group (step 8) are groups whose sequence of jobs is not determined (set $\bar{\Phi}^{\{\lambda\}}$) yet. Each group of this category is considered as a job, and their position is determined by model M_2 in such a way that it must be after the jobs in the first and second categories and according to sequence $\bar{\Phi}^{(\lambda)}$. In step 9, each time model M_2 is run, sequence Φ and the relative position of jobs from the previously defined group σ_λ are given as constant input to the new model M_2. In step 8, according to the sequences obtained from steps 3 and 9, the best possible schedule is found, and its corresponding objective function is the algorithm's output.

4.2. Group sequencing phase

In iteration λ ($\lambda \geq 2$) of model M_1, the relative position of $(\lambda - 1)$ groups, denoted by the symbol ς_p, is given as an input to model M_1. The relative position of group σ_λ is determined by implementing model M_1 among the groups whose relative position has already been fixed. That is, the output of model M_1 is the sequence of λ groups. Model M_1 reads as follows.

$$M_1 : Min\ TEC = \sum_{\tau\epsilon T} r_\tau . EG_\tau \tag{25}$$

Subject to: Constraints (45), (55) and (56)

$$F_p^{'m} - F_q^{'m} \geq -\beta Q'_{pq} + \pi_p^{'m},\ \forall m \in M; \forall\ p, q \in \Phi^{\{\lambda\}} : p < q; \tag{58}$$

$$F_q^{'m} - F_p^{'m} \geq \beta\left(Q'_{pq} - 1\right) + \pi_q^{'m}, \forall m \in M; \forall p, q \in \Phi^{\{\lambda\}} : p < q; \tag{59}$$

$$F_p^{'m} \geq F_p^{'m-1} + \pi_p^{'m}, \forall m \in M\backslash\{1\}; \forall p \in \Phi^{\{\lambda\}}; \tag{60}$$

$$F_p^{'\mu} \leq kd, \ \forall p \in \Phi^{\{\lambda\}}; \tag{61}$$

$$F_0^{'m} = 0, \forall m \in M; \tag{62}$$

$$Q'_{pq} = 1, if \ \lambda > 2; \forall \ p, q \in \Phi^{\{\lambda-1\}} : \ p < q, \varsigma_p < \varsigma_q; \tag{63}$$

$$Q'_{pq} = 0, if \ \lambda > 2; \forall \ p, q \in \Phi^{\{\lambda-1\}} : \ p < q, \varsigma_p < \varsigma_q; \tag{64}$$

$$\sum_{\tau \in \mathrm{T}} V_{p\tau}^{'m} = \pi_p^{'m}, \ \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \tag{65}$$

$$V_{p\tau}^{'m} \leq \pi_p^{'m} X_{p\tau}^{'m}, \ \forall \tau \in \mathrm{T}; \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \tag{66}$$

$$F_p^{'m} \geq ((\tau-1)d+1) - \beta\left(1 - X_{p\tau}^{'m}\right), \forall \tau \in \mathrm{T}\backslash\{1\}; \forall m \in M; \forall p \in \Phi^{\{\lambda\}} \tag{67}$$

$$F_p^{'m} - \pi_p^{'m} \leq (\tau d - 1) + \beta\left(1 - X_{p\tau}^{'m}\right), \forall \tau \in \mathrm{T}; \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \tag{68}$$

$$V_{p\tau}^{'m} \leq F_p^{'m} - (\tau-1)d + \beta\left(1 - X_{p\tau}^{'m}\right), \forall \tau \in \mathrm{T}\backslash\{1\}; \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \tag{69}$$

$$V_{p\tau}^{'m} \leq \tau d - \left(F_p^{'m} - \pi_p^{'m}\right) + \beta\left(1 - X_{p\tau}^{'m}\right), \forall \tau \in \mathrm{T}; \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \tag{70}$$

$$EG_\tau \geq \sum_{m \in M} \sum_{p \in \Phi^{\{\lambda\}}} pe_p^{'m} \ V_{p\tau}^{'m} - EB_\tau, \forall \tau \in \mathrm{T}; \tag{71}$$

$$EB_{\tau+1} \leq EB_\tau + EG_\tau + re_\tau - \sum_{m \in M} \sum_{p \in \Phi^{\{\lambda\}}} pe_p^{'m}.V_{p\tau}^{'m}, \forall \tau \in \mathrm{T}\backslash\{k\}; \tag{72}$$

$$0 \leq X_{p\tau}^{'m} \leq 1, \ \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \forall \tau \in \mathrm{T}; \tag{73}$$

$$Q'_{pq} \in \{0, 1\}, \ \forall p, q \in \Phi^{\{\lambda\}}; \tag{74}$$

$$F_p^{'m} \geq 0, \ \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \tag{75}$$

$$V_{p\tau}^{'m} \geq 0, \ \forall m \in M; \forall p \in \Phi^{\{\lambda\}}; \forall \tau \in \mathrm{T}; \tag{76}$$

Model M_1 is similar to the TIIM, except that in this model, only the sequencing of groups is addressed. In Constraints (58) and (59), the sequence of groups is determined. According to these Constraints, the completion time of each group must be equal to or larger than the completion time of all previously processed groups. Constraints (60) and (61) are similar to Constraints (36) and (38) in the TIIM, respectively. According to Constraints (62), the zero dummy group is placed at time zero. Constraints (63) and (64) are applied to the model if the number of groups is more than two ($\lambda > 2$). In this case, the positions of $(\lambda - 1)$ groups that have already been scheduled are fixed, and only the position of group σ_λ is investigated in this model. Constraints (65) to (70) are similar to Constraints (39) to (44) which have been modified by removing variables W_{ip} and replacing variables F_{ij}^m, $X_{pj\tau}^m$ and $V_{pj\tau}^m$ by the variables $F_p^{'m}$, $X_{p\tau}^{'m}$ and $V_{p\tau}^{''m}$ in the TIIM, respectively. Constraints (45) in the TIIM and Constraints (71) and (72) are used to calculate the energy consumption and energy stored in the battery. Constraints (55) and (56) in the TIIM and Constraints (73) to (76) determine the type of variable. Given that in model M_1 only the sequencing of groups is essential, so in Constraints (73), the variables $X_{p\tau}^{'m}$ are relaxed to reduce the CPU runtime of the model.

4.3. Job sequencing phase

In each iteration of model M_2, the sequence of groups obtained by model M_1 is considered as an input. Furthermore, the relative position $(\alpha - 1)$ jobs, denoted by $\xi_j^{\sigma_\lambda}$, is given to model M_2 as an input from the previous iteration to determine the sequence of jobs in group σ_λ. By implementation of model M_2, the relative position of job $\delta_\alpha^{\sigma_\lambda}$ is fixed among the jobs whose relative positions have already been determined. Therefore, the output of model M_2 is the relative order of α jobs belonging to group σ_λ. Finally, after the last iteration of model M_2, the sequences of jobs in all groups are determined. Model M_2 reads as follows.

$$M_2 : Min \ TEC = \sum_{\tau \in \mathrm{T}} r_\tau.EG_\tau \tag{25}$$

Subject to: Constraints (45), (55) and (56)

$$F_i^{''m} - F_h^{''m} \geq -\beta Q''_{ih} + \pi_i^{''m}, \ \forall m \in M; \forall \ i = 0, \ldots, \alpha + g - 1; \forall \ h = 1, \ldots, \alpha + g - 1 : i < h; \tag{77}$$

$$F_h^{''m} - F_i^{''m} \geq \beta\left(Q''_{ih} - 1\right) + \pi_h^{''m}, \forall m \in M; \forall \ i = 0, \ldots, \alpha + g - 1; \forall \ h = 1, \ldots, \alpha + g - 1 : i < h; \tag{78}$$

$$F_i^{''m} - F_{\lambda-1}^{''m} \geq s_{\sigma_{\lambda-1}\sigma_\lambda}^m + \pi_i^{''m}, \forall m \in M; \forall \ i = \lambda, \ldots, \alpha + g - 1; \tag{79}$$

$$F_i^{''m} \geq F_i^{''m-1} + \pi_i^{''m}, \forall m \in M\backslash\{1\}; \forall \ i = 1, \ldots, \alpha + g - 1; \tag{80}$$

$$F_i^{''\mu} \leq kd, \ \forall \ i = 1, \ldots, \alpha + g - 1; \tag{81}$$

$$F_0^{'m} = 0, \forall m \in M; \tag{82}$$

$$Q''_{ih} = 1, \ \forall \ i = 0, \ldots, \lambda - 1; \forall h = \lambda, \ldots, \alpha + g - 1; \tag{83}$$

$$Q''_{ih} = 1, \ \forall \ i = \lambda, \ldots, \lambda + \alpha - 1; \forall h = \lambda + \alpha, \ldots, \alpha + g - 1; \tag{84}$$

$$Q''_{ih} = 1, \ \forall \ i = 0, \ldots, \lambda - 1; \forall \ h = 0, \ldots, \lambda - 1 \ : i < h; \tag{85}$$

$$Q''_{ih} = 1, \ \forall i = \lambda + \alpha, \ldots, \alpha + g - 1; h = \lambda + \alpha, \ldots, \alpha + g - 1 \ : i < h; \tag{86}$$

$$Q''_{ih} = 1, if \ \alpha > 2; \forall \ i = \lambda, \ldots, \lambda + \alpha - 2; \forall h = \lambda, \ldots, \lambda + \alpha - 2 \ : i < h; \tag{87}$$

$$\sum\nolimits_{\tau \in \mathrm{T}} V_{i\tau}^{''m} = \pi_i^{''m}, \ \forall m \in M; \forall \ i = 1, \ldots, \alpha + g - 1; \tag{88}$$

$$V''^{m}_{i\tau} \le \pi''^{m}_{i} X''^{m}_{i\tau},\ \forall \tau \in \mathrm{T}; \forall m \in M; \forall\ i = 1, \ldots, \alpha + g - 1; \tag{89}$$

$$F''^{m}_{i} \ge ((\tau - 1)d + 1) - \beta\left(1 - X''^{m}_{i\tau}\right), \forall \tau \in \mathrm{T}\backslash\{1\}; \\ \forall m \in M; \forall\ i = 1, \ldots, \alpha + g - 1; \tag{90}$$

$$F''^{m}_{i} - \pi''^{m}_{i} \le (\tau d - 1) + \beta\left(1 - X''^{m}_{i\tau}\right),\ \forall \tau \in \mathrm{T}; \\ \forall m \in M; \forall\ i = 1, \ldots, \alpha + g - 1; \tag{91}$$

$$V''^{m}_{i\tau} \le F''^{m}_{i} - (\tau - 1)d + \beta\left(1 - X''^{m}_{i\tau}\right), \forall \tau \in \mathrm{T}\backslash\{1\}; \\ \forall m \in M; \forall\ i = 1, \ldots, \alpha + g - 1; \tag{92}$$

$$V''^{m}_{i\tau} \le \tau d - \left(F''^{m}_{i} - \pi''^{m}_{i}\right) + \beta\left(1 - X''^{m}_{i\tau}\right), \\ \forall \tau \in \mathrm{T}; \forall m \in M; \forall\ i = 1, \ldots, \alpha + g - 1; \tag{93}$$

$$EG_{\tau} \ge \sum\nolimits_{m \in M} \sum\nolimits_{i=1}^{\alpha+g-1} pe''^{m}_{i} V''^{m}_{i\tau} - EB_{\tau},\ \forall \tau \in \mathrm{T}; \tag{94}$$

$$EB_{\tau+1} \le EB_{\tau} + EG_{\tau} + re_{\tau} - \sum\nolimits_{m \in M} \sum\nolimits_{i=1}^{\alpha+g-1} pe''^{m}_{i} \cdot V''^{m}_{i\tau}, \\ \forall\ \tau \in \mathrm{T}\backslash\{k\}; \tag{95}$$

$$0 \le X''^{m}_{i\tau} \le 1,\ \forall m \in M; \forall i = 0, \ldots, \alpha + g - 1; \forall \tau \in \mathrm{T}; \tag{96}$$

$$Q''_{ih} \in \{0, 1\},\ \forall\ i = 0, \ldots, \alpha + g - 1; \forall\ h = 1, \ldots, \alpha + g - 1; \tag{97}$$

$$F''^{m}_{i} \ge 0,\ \forall m \in M; \forall\ i = 0, \ldots, \alpha + g - 1; \tag{98}$$

$$V''^{m}_{i\tau} \ge 0,\ \forall m \in M; \forall\ i = 0, \ldots, \alpha + g - 1; \forall \tau \in \mathrm{T}; \tag{99}$$

According to Constraints (77) and (78), the completion time of each job placed in a certain position must be equal to or larger than the completion time of the jobs processed in the previous positions. Constraints (79) pertain to the setup time between group σ_{λ} and its immediate precedent group. Constraints (80) and (81) are similar to Constraints (36) and (38) in the TIIM, respectively. Constraints (82) state that on each machine, the completion time of the job placed at the dummy position is zero. According to Constraints (83), the groups in sequence Φ that precede group σ_{λ} must be processed before the jobs of group σ_{λ}, as well as before the jobs of groups in sequence $\bar{\Phi}^{(\sigma_{\lambda})}$. Constraints (84) also denote that the jobs of group σ_{λ} must be processed before the groups involved in sequence $\bar{\Phi}^{(\sigma_{\lambda})}$. Constraints (85) and (86) determine the sequence of groups $\Phi^{\{\lambda-1\}}$ and $\bar{\Phi}^{\{\lambda\}}$ according to sequence $\Phi^{(\lambda-1)}$ and $\bar{\Phi}^{(\lambda)}$, respectively. Constraints (87) are applied to model M_2 if the number of jobs in group σ_{λ} is more than two jobs ($\alpha > 2$). In these Constraints, the relative positions of $(\alpha - 1)$ jobs, which have already been determined by model M_2, are fixed, and only the position of job $\delta_{\alpha}^{\sigma_{\lambda}}$ is explored. Constraints (88) to (95) perform similar to Constraints (65) to (72) in model M_1. Moreover, Constraints (45) in the TIIM are also used in this model. Constraints (55) and (56) in the TIIM and Constraints (96) to (99) show the type of variables. Since in model M_2 only sequencing of jobs within each group is significant, in Constraints (96) variables $X''^{m}_{i\tau}$ are relaxed to reduce the CPU runtime of the model.

4.4. Scheduling phase

Model M_3 is related to the scheduling phase of groups and jobs within each group, and it is implemented only once. In this model, the sequence of groups obtained by model M_1 and the sequence of jobs in each group obtained by model M_2 are the inputs of the model. Model M_3 gives rise to a high-quality schedule to reduce the total cost of electricity purchased from the grid. Model M_3 determines the start times of all jobs and groups along with the total cost of electricity. Model M_3 can be formulated as follows.

$$M_3 : Min\ TEC = \sum_{\tau \in \mathrm{T}} r_{\tau}.EG_{\tau} \tag{25}$$

Subject to: Constraints (39), (40), (45), (51), (54), (55) and (56)

$$F'''^{m}_{pj} - F'''^{m}_{p(j-1)} \ge \pi^{m}_{pj},\ \forall m \in M; \forall p \in \Phi\backslash\{0\}; \\ \forall j \in \Gamma_p :\ if\ \xi^{p}_{j} = \xi^{p}_{j-1} + 1; \tag{100}$$

$$F'''^{m}_{pj} - F'''^{m}_{ql} \ge \pi^{m}_{pj} + s^{m}_{qp},\ \forall m \in M; \forall q \in \Phi; \forall p \in \Phi\backslash\{0\} : \\ if\, \varsigma_p = \varsigma_q + 1; \forall j \in \Gamma_p : if\ \xi^{p}_{j} = 1; \forall l \in \Gamma_q : \\ if\ \xi^{q}_{l} = n_q; \tag{101}$$

$$F'''^{m}_{pj} - F'''^{m-1}_{pj} \ge \pi^{m}_{pj},\ \forall m \in M\backslash\{1\}; \forall p \in \Phi\backslash\{0\}; \forall j \in \Gamma_p; \tag{102}$$

$$F'''^{\mu}_{pj} \le kd,\ \forall p \in \Phi\backslash\{0\}; \forall j \in \Gamma_p; \tag{103}$$

$$F''^{m}_{00} = 0, \forall m \in M; \tag{104}$$

$$F'''^{m}_{pj} \ge ((\tau - 1)d + 1) - \beta\left(1 - X^{m}_{pj\tau}\right), \forall \tau \in \mathrm{T}\backslash\{1\}; \\ \forall m \in M;\ p \in \Phi\backslash\{0\}; \forall j \in \Gamma_p; \tag{105}$$

$$F'''^{m}_{pj} - \pi^{m}_{pj} \le (\tau * d - 1) + \beta\left(1 - X^{m}_{pj\tau}\right), \forall \tau \in \mathrm{T}; \\ \forall m \in M; \forall p \in \Phi\backslash\{0\}; \forall j \in \Gamma_p; \tag{106}$$

$$V^{m}_{pj\tau} \le F'''^{m}_{pj} - (\tau - 1)d + \beta\left(1 - X^{m}_{pj\tau}\right), \forall \tau \in \mathrm{T}\backslash\{1\}; \\ \forall m \in M; \forall p \in \Phi\backslash\{0\}; \forall j \in \Gamma_p; \tag{107}$$

$$V^{m}_{pj\tau} \le \tau d - \left(F'''^{m}_{pj} - \pi^{m}_{pj}\right) \\ + \beta\left(1 - X^{m}_{pj\tau}\right), \forall \tau \in \mathrm{T}; \forall m \in M; \forall p \in \Phi\backslash\{0\}; \forall j \in \Gamma_p; \tag{108}$$

$$F'''^{m}_{pj} \ge 0, \forall m \in M; \forall p \in \Phi\backslash\{0\}; \forall j \in \Gamma_p; \tag{109}$$

Constraints (100) determine the completion time of the jobs in each group according to the sequence of jobs in that group, determined in step 9 of the DBHA. Based on the sequence of groups determined in step 3 of the DBHA, Constraints (101) determine the completion time of each group that must be after the completion time of the last job of the previous group and the corresponding setup time between these two groups. Constraints (102) and (103) are similar to Constraints (36) and (38) in the TIIM. According to Constraints (104), the dummy job is placed at time zero. Constraints (105) to (108) function like Constraints (67) to (70) of model M_1. Constraints (39) and (40) and Constraints (45) are also included in model M_3, with the difference that $p \in G^*$ and $j \in N$ are replaced by $p \in \Phi$ and $j \in \Gamma_p$, respectively. Constraints (51) and (54) to (56) of the TIIM and Constraint (109) indicate the type of variables. In addition, in these constraints, $p \in G^*$ is replaced by $p \in \Phi$, and $j \in N$ is replaced by $j \in \Gamma_p$.

Table 5
TOU pricing

Season	Electricity price ($/kWh)	Time period	Peak hours
Summer	0.1762	off-peak load	00-9, 22-24
	0.2478	mid-peak load	9-12, 18-22
	0.5446	on-peak load	12-18
Winter	0.1758	off-peak load	22-9
	0.1940	on-peak load	9-22

5. Computational experiments

In this section, we evaluate the performance of the TUIM, TIIM and DBHA. We also analyze the sensitivity of various parameters and evaluate the economic performance of the research. The DBHA and developed MIP models were implemented in Microsoft Visual Studio 2015 using ILOG CPLEX 12.6 and run on a computer with Intel (R) Core (TM) i7-6800k 3.4GHz processor and 48 GB memory.

5.1. Data set generation

To evaluate the performance of the models and algorithm, two groups of small-size and medium-size instances were developed. The number of groups in the small-size instances is 3, 4, or 5 and in the medium-size instances is 6, 7, or 8. In addition, the number of jobs in each group is 3, 4, or 5 for small-size instances and 6, 7, or 8 for medium-size instances. Furthermore, in each instance, the number of jobs in each group was considered equal. The number of machines for both small-size and medium-size instances is 3, 4, 5. For each combination of different values of the number of groups, jobs, and machines, ten instances were generated, resulting in a total of 540 instances. The other parameters for both small- and medium-size instances are developed randomly.

To consider a scheduling horizon, we make use of the NEH algorithm developed by Nawaz et al. (1983). To do so, we ignore power consumption of jobs and setup times between them, and we also consider each group as a job in such a way that the processing time of each group equals the summation of processing times of jobs belonging to it. Next, we determine the sequence of groups in the way that the makespan is minimized. If we were to consider the resultant makespan of the NEH algorithm, shown by C_{max}^{NEH}, as the scheduling horizon, we would, of course, have a feasible solution, but it might lead to a high cost of purchasing electricity from the grid. To alleviate this problem, we let $T_{max}(or\ kd) = \eta C_{max}^{NEH}$ as the length of the scheduling horizon in all developed solution approaches. Using a fine-tuning approach, we found that $\eta = 1.5$ is an appropriate value for the problem at hand. Furthermore, we assume that the scheduling horizon starts at 8 a.m.

According to Schaller et al. (2000), the processing time follows a uniform distribution in the range of $\{1, \ldots, 10\}$ and the setup time of each group is also a discrete uniform distribution of $\{1, \ldots, 20\}$. The electricity price based on TOU for different electricity suppliers varies in the summer (spring and summer) and winter (autumn and winter). According to Zhang et al. (2017), the electricity price in the summer and winter in California is shown in Table 5, and we chose to generate our instances based on the electricity price in the summer. The power consumed by each machine follows a uniform distribution at intervals of {5kW,...,20kW} during the processing of jobs. Battery capacity is also considered to be 25 kWh.

Table 6 shows the electrical energy produced by PV panels at different hours, assuming that the power of the PV panels installed in the factory is equal to 25 kWp and the energy wasted in the whole system is 14%. These data are based on the average energy produced at different times of the day in July 2015 in California, extracted from European Commission (2020). Since the solar energy production reaches zero at night, the energy production is considered to be zero from 8 p.m. to 7 a.m., and in the other hours is according to Table 6.

5.2. Comparative computational results

In this section, using the generated instances, we evaluate the comparative performance of the two developed models and the DBHA. To examine more closely, we consider the TIIM and the DBHA in four different scenarios where the length of time intervals varies from one scenario to another. In the first scenario, we consider $d = 60$ minutes, and in the second, third, and fourth scenarios, we factor in $d = 30,\ 15,$ and 1 minute(s), respectively. Furthermore, we consider the relaxed version of the TIIM model with $d = 1$ minute as a valid lower bound (LB) for all developed models. The average CPU runtime (seconds), the number of instances solved optimally, and the average deviation (AD) of objective function values from the LBs are used to evaluate the performance of the developed solution approaches.

It is worth mentioning again that we apply the term "*local optimal*" for the optimal solutions obtained by the TIIM with $d > 1$. We also make use of the term "*global optimal*" for the optimal solutions obtained by the TUIM or the TIIM with $d = 1$. In addition, a one-hour time limit is considered as the CPU runtime for all implementations.

Table 7 shows the performance of the TUIM and the different scenarios of the TIIM in small-size instances based on two criteria: the number of instances solved locally or globally optimal in the given time limit and the average CPU runtime of these instances. The symbol "−−" indicates that no instances could be solved optimally, and hence there is no report of CPU runtime for such instances.

According to the results of Table 7, the TUIM finds an optimal solution for only a small number of instances, whereas the TIIM solves locally optimal all instances in three scenarios $d = 60$, $d = 30$, and $d = 15$. The TIIM in scenario $d = 1$ also finds the global optimal solution in most instances, but with the increase of the number of jobs, groups and machines, the number of instances which can be optimally solved declines. For example, when $\mu = 4$, $g = 5$, and $n_p = 5$, none of the instances in this scenario can be solved optimally. By comparing the instances that are solved optimally, it can be seen that the average CPU runtime is much less in the TIIM than the TUIM, even in the fourth scenario of the TIIM, where its accuracy is equal to that of the TUIM, and the solutions of both are globally optimal. It should be noted that the number of instances that can be solved optimally using the TIIM in the fourth scenario is much higher than the TUIM, and also, the average CPU runtime is significantly less than the TUIM. This shows the superiority of the TIIM over the TUIM.

By examining the different scenarios of the TIIM, it can be concluded that in each scenario of the TIIM, with the increase of the number of groups, jobs, and machines, the CPU runtime also increases. In addition, by decreasing d from 60 minutes to 1 minute, the average CPU runtime rises in all scenarios of the TIIM. In other words, the more accuracy of the model, the more CPU runtime.

Table 8, shown in the Appendix, compares the performance of the TIIM and DBHA and indicates that for some smaller-size instances, the TIIM outperforms the DBHA but for larger-size ones, the DBHA shows better performance.

Given the one-hour time limit, three different modes for solving the models may occur: (a) the model finds the optimal solution, (b) the model finds a feasible solution, and (c) the model fails to find any feasible solution. Table 9 shows the comparative performance of the TIIM and the DBHA based on the AD from the LB. It is worth mentioning that the AD is calculated only for cases that at least a feasible solution has been obtained. It should be noted that

Table 6
Electricity produced by PV panels

Hours	7-8	8-9	9-10	10-11	11-12	12-13	13-14	14-15	15-16	16-17	17-18	18-19	19-20
Generated electricity (kWh)	0.5	4	8.5	12.7	14.8	16.8	17.5	16.8	15	11.6	8	3.3	0.5

Table 7
The comparative performance of the TUIM and TIIM for small-size instances

μ	g	n_p	The number of instances solved locally or globally optimal					The average CPU runtime (seconds) of the instances solved locally or globally optimal				
			TUIM	TIIM-60min	TIIM-30min	TIIM-15min	TIIM-1min	TUIM	TIIM-60min	TIIM-30min	TIIM-15min	TIIM-1min
3	3	3	2	10	10	10	10	2623.9	0.1	0.3	0.7	58.1
		4	0	10	10	10	10	––	0.3	0.9	1.7	197.8
		5	1	10	10	10	10	2492.5	0.5	0.9	3.3	764.2
	4	3	0	10	10	10	10	––	0.4	0.6	2.0	192.5
		4	0	10	10	10	10	––	0.9	0.9	10.7	929.5
		5	0	10	10	10	9	––	3.5	3.7	25.2	1067.0
	5	3	0	10	10	10	10	––	0.7	2.0	9.1	732.6
		4	0	10	10	10	8	––	3.6	6.2	38.4	1437.3
		5	0	10	10	10	1	––	9.1	12.8	111.0	2597.4
4	3	3	0	10	10	10	10	––	0.3	0.4	1.1	152.6
		4	0	10	10	10	10	––	0.6	1.1	3.6	610.7
		5	0	10	10	10	10	––	1.3	2.7	8.4	1182.6
	4	3	0	10	10	10	10	––	0.8	0.9	4.9	915.0
		4	0	10	10	10	6	––	2.1	2.2	8.0	1237.4
		5	0	10	10	10	4	––	6.2	7.5	26.5	2324.6
	5	3	0	10	10	10	6	––	2.8	3.9	20.0	2155.7
		4	0	10	10	10	2	––	8.9	16.8	65.4	3099.3
		5	0	10	10	10	0	––	28.1	58.6	260.0	––
5	3	3	0	10	10	10	10	––	0.5	0.6	1.7	342.4
		4	0	10	10	10	6	––	0.8	1.3	5.6	1209.2
		5	0	10	10	10	3	––	2.8	2.8	15.7	1645.4
	4	3	0	10	10	10	9	––	1.2	1.7	5.7	924.6
		4	0	10	10	10	2	––	4.2	5.2	19.5	1968.0
		5	0	10	10	10	0	––	9.1	29.8	68.7	––
	5	3	0	10	10	10	1	––	5.6	8.4	39.6	3002.2
		4	0	10	10	10	0	––	22.3	35.5	159.9	––
		5	0	10	10	10	0	––	55.9	216.7	623.9	––

Table 9
The comparative performance of the TIIM and the DBHA based on the AD for small-size instances

μ	g	n_p	TUIM	TIIM				DBHA			
				60min	30min	15min	1min	60min	30min	15min	1min
3	3	3	0.24 (10)	2.46	1.33	0.76	0.21	2.46	1.33	0.76	0.22
		4	0.51 (9)	2.72	1.41	0.82	0.27	2.72	1.41	0.82	0.28
		5	0.42 (8)	2.59	1.21	0.67	0.30	2.59	1.21	0.67	0.31
	4	3	0.44 (8)	2.35	1.15	0.64	0.22	2.35	1.15	0.64	0.25
		4	1.25 (3)	3.05	1.59	0.86	0.26	3.05	1.60	0.88	0.32
		5	0.91 (2)	2.90	1.39	0.79	0.44	2.90	1.39	0.79	0.45
	5	3	0.88 (3)	2.03	0.67	0.37	0.30	2.03	0.67	0.37	0.30
		4	––	2.33	0.95	0.56	0.56	2.33	0.96	0.56	0.45
		5	––	3.37	1.86	1.16	7.26	3.37	1.86	1.16	0.62
4	3	3	0.27 (10)	2.62	1.33	0.75	0.21	2.62	1.33	0.75	0.22
		4	0.64 (8)	2.62	1.16	0.63	0.34	2.62	1.16	0.63	0.35
		5	0.42(2)	3.11	1.60	0.92	0.35	3.11	1.60	0.92	0.38
	4	3	0.42 (3)	3.22	1.78	1.05	0.36	3.22	1.78	1.05	0.37
		4	––	3.27	1.76	1.00	0.58	3.27	1.76	1.01	0.39
		5	––	3.31	1.80	1.05	2.36	3.31	1.80	1.06	0.51
	5	3	––	3.15	1.64	0.92	1.48	3.15	1.64	0.92	0.45
		4	––	3.48	1.96	1.20	2.68	3.48	1.96	1.20	0.62
		5	––	3.27	1.90	1.21	9.30	3.28	1.90	1.22	0.63
5	3	3	0.46 (6)	2.83	1.48	0.85	0.31	2.83	1.48	0.85	0.36
		4	2.29 (2)	3.32	1.82	1.08	0.47	3.32	1.82	1.08	0.43
		5	––	3.43	1.90	1.13	7.06	3.43	1.90	1.13	0.48
	4	3	0.59 (1)	3.20	1.72	1.06	0.42	3.20	1.72	1.07	0.49
		4	––	3.44	1.91	1.15	4.21	3.44	1.91	1.16	0.57
		5	––	3.31	1.86	1.13	7.40	3.31	1.86	1.14	0.48
	5	3	––	3.38	1.85	1.10	3.53	3.38	1.86	1.10	0.53
		4	––	3.47	2.00	1.27	8.34	3.47	2.00	1.27	0.63
		5	––	3.25	1.86	1.20	15.37	3.27	1.88	1.21	0.65

Table 12
The comparative performance of the TIIM and the DBHA based on the AD values of medium-size instances

μ	g	n_p	TIIM 60min	30min	15min	1min	DBHA 60min	30min	15min	1min
3	6	6	2.44(10)	1.42 (10)	0.91 (10)	12.84 (10)	2.44	1.42	0.91	0.47
		7	1.90 (10)	1.15 (10)	0.77 (10)	12.66 (10)	2.02	1.22	0.77	0.44
		8	2.47 (10)	0.96(10)	0.78 (10)	13.01)10)	1.73	1.03	0.79	0.54
	7	6	1.94 (10)	1.17 (10)	0.78 (10)	13.00 (10)	1.94	1.17	0.78	1.64
		7	2.45 (10)	1.00 (10)	0.77 (10)	12.68 (10)	1.64	1.13	0.73	2.63
		8	3.77 (10)	2.62 (10)	2.10 (10)	14.07 (6)	2.94	1.86	1.83	3.64
	8	6	1.51 (10)	1.42 (10)	1.09 (10)	10.68 (7)	1.51	1.09	0.91	2.16
		7	3.44 (10)	1.80 (10)	1.55 (10)	11.30 (4)	1.55	1.49	1.4	3.53
		8	8.44 (10)	8.78 (10)	5.33 (10)	––	3.79	3.68	3.65	4.86
4	6	6	1.97(10)	1.22 (10)	0.87 (10)	16.05 (10)	2.05	1.31	0.84	6.85
		7	2.78 (10)	1.49 (10)	1.47 (10)	13.83 (7)	2.49	1.86	1.33	5.48
		8	4.59 (10)	3.58 (10)	3.55 (10)	18.80 (5)	3.93	3.74	3.48	9.79
	7	6	3.08 (10)	2.56 (10)	2.40 (10)	14.66 (5)	2.84	2.29	1.98	5.08
		7	7.76 (10)	6.33 (10)	5.65 (10)	18.10 (2)	3.99	3.96	3.91	6.92
		8	12.42 (10)	10.25 (10)	7.98 (10)	––	6.35	6.33	6.26	15.2
	8	6	6.08 (10)	4.86 (10)	4.10 (10)	20.74 (3)	3.67	3.66	3.41	9.97
		7	17.15 (10)	12.43 (10)	7.50 (10)	––	6.16	6.11	6.09	9.94
		8	27.50 (10)	15.69 (10)	7.91 (10)	––	6.72	6.48	6.42	18.8
5	6	6	3.27 (10)	2.85 (10)	2.59 (10)	18.14 (10)	3.5	2.61	2.26	8.82
		7	5.47 (10)	5.23 (10)	4.38 (9)	22.95 (6)	4.68	4.64	4.16	15.52
		8	13.97 (10)	8.88 (10)	7.97 (8)	––	7.27	7.25	7.2	13.52
	7	6	10.78 (10)	7.06 (10)	4.72 (8)	23.83 (5)	5.19	4.96	4.57	9.24
		7	15.83 (10)	13.09 (9)	12.50 (7)	––	8.21	8.1	8.04	17.36
		8	22.53 (10)	15.81 (9)	15.16 (8)	––	9.51	9.4	9.34	30.47
	8	6	24.14 (10)	22.00 (10)	14.87 (8)	––	7.65	7.52	7.47	14.71
		7	30.5 (10)	15.42 (9)	15.09 (7)	––	8.5	8.36	8.26	24.68
		8	32.78 (10)	18.57 (8)	12.01 (6)	––	8.5	8.36	8.15	41.43

for all (some) instances that the TIIM (TUIM) fails to find their optimal solutions, it has been able to find a feasible solution for each. Moreover, the numbers inside parentheses indicate the number of instances that the TUIM could find at least a feasible solution, and the AD values have been calculated only based on these instances.

Table 9 indicates that in most cases, the AD values get smaller from left to right for both the TIIM and the DBHA, implying smaller values of d lead to more high-quality solutions. Although it is supposed that the TIIM with $d = 1$ always leads to more elite solutions, it does not hold in all cases because the TIIM fails to solve some instances optimally in the given time limit. Consequently, the AD surges conspicuously in some cells related to the TIIM with $d = 1$, particularly where $\mu = 4,\ 5$ and $n_p = 5$. This implies that increasing the length of time intervals might contribute to better performance of the model, mainly when we deal with larger sizes of instances. Considering scenarios $d = 30$ and 60 minutes, Table 9 indicates that the DBHA and the TIIM have almost identical performances in terms of AD measure. The columns related to scenario $d = 15$ indicate a bit better performance of the TIIM. For scenario $d = 1$, the TIIM often outperforms the DBHA in smaller-size instances, but this superiority gets reversed if the size of instances increases.

Tables 10 and 11, presented in the Appendix, compare the performance of the TIIM and the DBHA under different scenarios for medium-size instances.

Table 12 compares the different TIIM and the DBHA scenarios for medium-size instances based on the average deviation from the LB. The numbers inside parentheses shows the number of medium-size instances for which the TIIM can find a feasible solution (optimal or non-optimal). As can be seen, in all scenarios, this measure declines if the size of instances increases. In Table 12, the AD values are calculated only for the instances that at least a feasible solution is found. This table reveals that the DBHA has a better performance than the TIIM. Similar to Table 9, the scenarios corresponding to $d = 15$ leads to the best performance of the TIIM. In contrast to Table 9, the best performance of the DBHA for medium-size instances is attained with the scenarios corresponding to $d = 15$.

Table 13
The statistical comparison of the DBHA and the TIIM based on small-size instances

d	1	15	30	60
stat-value	69	0	0	3
critical-value	69	92	92	92
Decision	Accept H_0	Reject H_0	Reject H_0	Reject H_0

5.3. Statistical analyses

In this section, we aim to compare the performance of the TIIM and the DBHA from a statistical viewpoint. We exclude the TUIM because it showed a poor performance conspicuously. To compare the performances of the TIIM and the DBHA fairly, we need to define a new measurement to capture both solution quality and CPU runtime. To this end, we introduce $\rho = \text{AD}\times\text{Runtime}$ and utilize the non-parametric Wilcoxon signed-rank test. Furthermore, we have to consider only instances that can be solved by both solution methods. For each statistical assessment, the null hypothesis indicates that there is no significant difference between performances of the two solution approaches, whereas the one-side alternative hypothesis implies that the DBHA has a smaller average value than the TIIM. Table 13 indicates the results of this comparison for the small-size instances and different values of d. The stat-value shows the values of the test statistic and has to be equal to or larger than the critical value to accept the null hypothesis. We determine the critical value based on the sample size and $\alpha = 0.01$, where α indicates the value of type-I error. As can be seen from Table 13, for small instances and $d = 1$, we cannot reject the null hypothesis, but for $d = 15,\ 30$, and 60, we can conclude the DBHA outperforms the TIIM.

Table 14
The statistical comparison of the DBHA and the TIIM based on medium-size instances

d	15	30	60
Stat-value	0	0	3
Critical-value	3	5	5
Decision	Reject $\boldsymbol{H}_0$	Reject $\boldsymbol{H}_0$	Reject $\boldsymbol{H}_0$

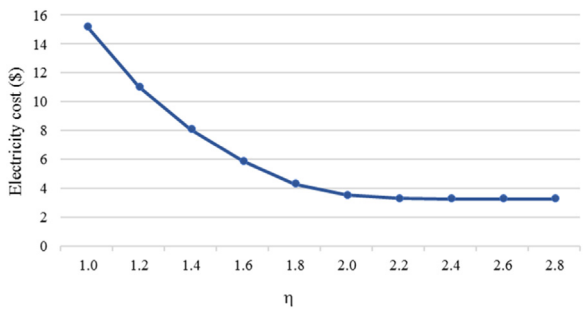

Fig. 5. The impact of scheduling horizon on the daily cost of electricity.

Likewise, we can construct Table 14 in which the DBHA and the TIIM are statistically compared based on medium-size instances. We exclude the case of d=1 because the TIIM fails to solve it. According to Table 14, the DBHA performs better than the TIIM.

5.4. Sensitivity analyses and evaluation of economic performance

In this section, we analyze the impact of parameters ca and re_t and η on the electricity costs. To do so, we consider an instance including 5 machines, 5 groups and 5 jobs in each group, and we apply the TIIM with $d = 15$ to find optimal solutions. The cost of electricity in the summer and winter is considered according to Table 5. Furthermore, we consider 261 working days in a year and assumed the energy produced for one year is also related to 2015 in California, extracted from European Commission (2020).

Fig. 5 shows the effect of the scheduling horizon on daily electricity costs. According to this figure, the cost of electricity consumption decreases with the extension of the scheduling horizon, but the downward trend of costs is stopped for $\eta > 2$. This is due to the limited energy generated by the panels and the lack of renewable energy at night. If we choose tight scheduling horizons ($\eta = 1$), we might reach low-quality feasible solutions in terms of electricity cost. On the other hand, loose scheduling horizons ($\eta \geq 1.5$) result in schedules with sparse jobs across the scheduling horizon. It is worth mentioning that for all instance examined in Fig. 5, we obtained the makespans corresponding to optimal solutions. In all cases, the makespan was equal to the time horizon. In other words, in comparison with values of C_{max}^{NEH}, makespans deteriorate $(\eta - 1) \times 100$ percent on average under the energy-aware scheduling approach.

Fig. 6 shows the changes in the annual electricity costs for different battery capacities and PV panel power. If we consider the battery capacity as a constant value, the cost of electricity consumption will decrease with the increase of the power of PV panels. In addition, if we consider the power of PV panels as a constant value, the annual cost of electricity will decrease with the increase of the battery capacity.

To calculate electricity cost savings in various months, two modes have been considered: (1) no use of renewable energy and (2) the possibility of using renewable energy. For the first case, the model was implemented without using renewable energy to reduce electricity costs. In the second case, the power of the PV panels and the battery capacity were considered 25 kWp and 25 kWh, respectively. Fig. 7 shows cost savings resulting from the difference in the cost of purchased electricity in these two cases over different months of summer and winter. As shown in Fig. 7, the cost savings in summer (May-October) are higher than in winter.

Figs. 8 and 9 analyze the economic impact of ca and re_t. For these analyses, the electricity costs of the previous instance were considered in the annual electricity costs. In addition, to obtain electricity cost savings due to the use of PV panels and the battery, the instance was solved in two modes, i.e., with and without

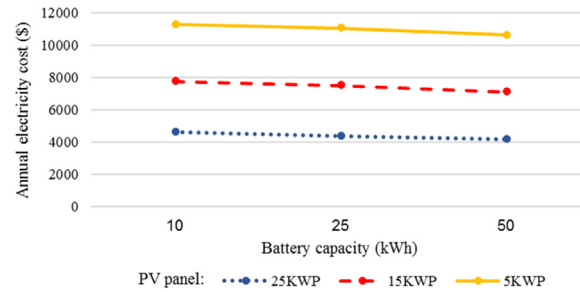

Fig. 6. The impact of the change in the battery cap.

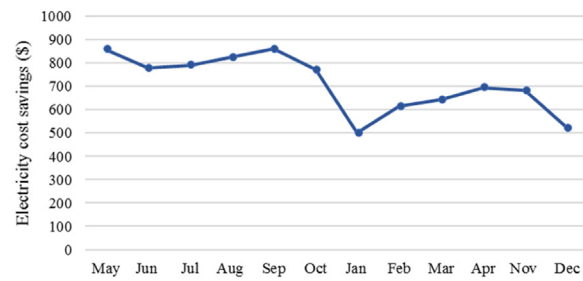

Fig. 7. The cost savings of electricity costs in summer and winter.

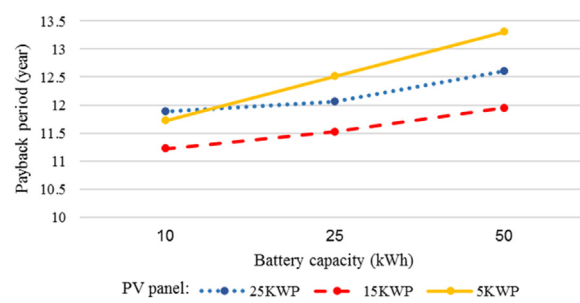

Fig. 8. The payback period on various battery capacities and panel powers.

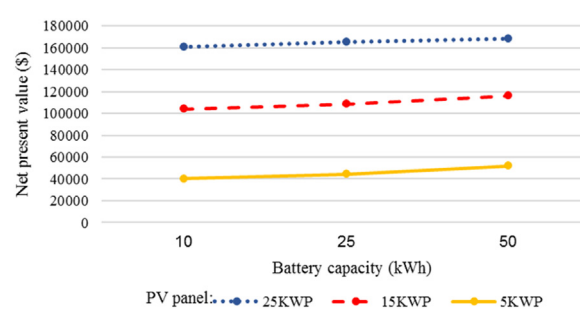

Fig. 9. The NPV for different battery capacity and panel power.

Table 15
PV panels installation and maintenance cost

Technology type	Installation cost($/kWp)	Fixed operation & maintenance cost ($/(kWp.year))	Lifespan(year)
<10kWp	3897	21	33
10-100 kWp	3463	19	33

considering the possibility of using renewable energy and battery storage system. Table 15 shows the costs associated with purchasing and installing PV panels, maintenance as well as the lifespan of the panel. This information was extracted from reference NREL transforming energy (2020). The cost of the used battery is also 300 $/kWh.

For simplification, we assume that we have to schedule the same instance on all working days of a year. On the ground of this assumption and based on the information provided in Table 15, Fig. 8 shows the payback period (year) for different powers of the PV panels and battery capacity. According to this figure, the shortest payback period is gained when a panel of 15 kWp and a 10 kWh battery are used. Furthermore, the longest payback period is obtained when a panel of 5 kWp and a 50 kWh battery are utilized.

Fig. 9 shows the net present value (NPV) for different powers of PV panels and battery capacities. Based on Zhang et al. (2017), the interest rate is considered 0.04 to calculate the NPV. If the NPV is negative, it means that the return on the investment is less than the total cost of installation, purchase, and maintenance of the battery and PV panels. Furthermore, if the NPV is positive, it signifies the investment has financial benefits, and the return on the investment is higher than the total cost of installation, purchase, and maintenance of the battery and PV panels. According to Fig. 9, all various sizes of battery capacity and power of the PV panels are economical and can be used. With the increase of the panel power and battery capacity, the NPV rises as well so that the largest NPV is related to the case that the power of the PV panels is 25 kWp and the battery capacity is 50 kWh. Furthermore, the effect of the power of the PV panels on the NPV is greater than the effect of the battery capacity.

6. Summary, conclusions, and future outlook

In this research, the flow shop scheduling problem was investigated by assuming group scheduling and considering electricity costs and the possibility of using solar energy. Two mixed-integer linear programming models in conjunction with a decomposition-based heuristic algorithm were developed to solve the problem. In the structure of the first model, the variables are defined based on the unit of time (minutes), but in the second model, the scheduling horizon is divided into some time intervals of equal length, and the variables are defined according to these time intervals. The computational results indicate that the second model has more flexibility considering the possibility of changing the length of time intervals and also consumes much fewer CPU runtimes than the first one. The best performance of the second model is achieved when the length of time intervals is set to 15 minutes though this might not lead to globally optimal solutions. Given the computational performance of the heuristic algorithm, this algorithm has the best performance among all three developed solution approaches. Furthermore, the best performances of the heuristic algorithm for small-size and medium-size instances are obtained when the length of time intervals is set to 1 and 15 minutes, respectively.

Sensitivity analysis examined the effect of PV panel power, battery capacity, and scheduling horizon on increasing or decreasing the costs of electricity consumption. According to the results, with the increase of the scheduling horizon, the costs of electricity consumption decreases, but after a while, with the increase of the scheduling horizon, the decreasing trend of costs stops and an increase in the scheduling horizon will no longer have an effect on reducing electricity costs. By examining the savings in electricity costs in summer and winter, it can be concluded that the reduction in electricity costs in summer is greater than in winter. In addition, the duration of the payback period and the net present value of capital were examined in different power of panels and battery capacity. These results can be economically valuable for identifying the proper panels and battery capacity.

As a suggestion for future research, the problem assumptions can be extended by considering non-permutation flow shop scheduling problems and using other production environments such as the flexible flow shop or the job shop. The possibility of changing the speed of machines, machine breakdowns, or the maintenance process would appear to be interesting research topics. Furthermore, we did not consider the labor costs in our work, whereas it can be expensive in many countries, particularly during out-of-working hours. Thus, we suggest considering this cost in the problem for future research. In addition to these, developing more efficient exact and heuristic algorithms that can solve large-size instances is fascinating and challenging research.

Appendix

Tables 8, 10, 11

Table 8
The average CPU runtimes (seconds) of the DBHA for small-size instances

μ	g	n_p	DBHA-60min	DBHA -30min	DBHA -15min	DBHA -1min
3	3	3	0.26	0.16	0.26	3.53
		4	0.25	0.30	0.49	29.70
		5	0.32	0.43	0.83	35.31
	4	3	0.22	0.31	0.52	4.43
		4	0.32	0.51	1.00	175.50
		5	0.49	0.86	1.59	195.47
	5	3	0.34	0.50	0.99	44.35
		4	0.48	0.81	1.46	212.64
		5	0.69	1.14	1.87	681.12
4	3	3	0.28	0.27	0.45	21.14
		4	0.30	0.47	0.90	55.26
		5	0.36	0.65	1.13	170.54
	4	3	0.25	0.44	0.65	96.60
		4	0.37	0.77	1.19	642.67
		5	0.63	1.19	1.89	1044.51
	5	3	0.36	0.72	1.02	783.66
		4	0.57	1.02	1.80	1361.53
		5	1.05	2.11	3.33	1913.92
5	3	3	0.32	0.31	0.58	59.20
		4	0.33	0.59	0.92	775.03
		5	0.51	0.96	1.60	1630.60
	4	3	0.28	0.48	0.85	185.45
		4	0.49	0.95	1.76	1113.41
		5	0.83	1.46	2.91	2683.74
	5	3	0.50	0.82	1.60	1059.65
		4	0.86	1.27	2.65	2660.76
		5	1.43	2.19	4.40	2821.72

Table 10
The performance of the TIIM for medium-size instances

μ	g	n_p	The number of instances solved locally or globally optimal				The average CPU runtime (seconds) of instances solved locally or globally optimal			
			TIIM-60min	TIIM-30min	TIIM-15min	TIIM-1min	TIIM-60min	TIIM-30min	TIIM-15min	TIIM-1min
3	6	6	10	10	10	0	92.7	350.7	599.1	––
		7	10	10	10	0	407.8	969.3	860.6	––
		8	6	6	4	0	519.4	1167.4	1541.6	––
	7	6	10	10	10	0	406.1	797.2	1379.5	––
		7	8	7	6	0	1957.5	2097.4	2704.9	––
		8	3	2	2	0	3252.5	3288.0	3294.4	––
	8	6	8	8	6	0	1775.6	1917.3	3114.9	––
		7	0	0	0	0	––	––	––	––
		8	0	0	0	0	––	––	––	––
4	6	6	9	8	6	0	345.5	972.3	1092.6	––
		7	3	2	2	0	2667.1	2681.6	2992.1	––
		8	0	0	0	0	––	––	––	––
	7	6	2	2	0	0	2600.3	3238.9	––	––

Table 11
The average CPU runtime (in seconds) of the DBHA for medium-size instances

μ	g	n_p	DBHA-60min	DBHA -30min	DBHA -15min	DBHA -1min
3	6	6	1.75	2.54	6.73	2259.79
		7	2.02	3.54	7.86	2667.10
		8	2.74	4.58	12.86	2965.32
	7	6	2.23	3.24	7.94	3057.56
		7	2.72	4.09	11.24	3217.20
		8	3.49	5.74	12.53	3600
	8	6	2.68	4.39	12.13	3423.04
		7	3.52	6.26	13.85	3273.72
		8	3.73	6.83	10.43	3600
4	6	6	2.31	3.86	9.42	3121.76
		7	2.61	4.95	11.79	3600
		8	3.66	6.01	16.82	3600
	7	6	2.60	5.09	10.70	3453.47
		7	3.78	6.60	17.09	3600
		8	4.20	7.42	21.12	3600
	8	6	3.48	5.80	18.49	3600
		7	3.71	7.43	19.78	3600
		8	4.67	10.20	34.08	3600
5	6	6	2.62	5.22	15.17	3600
		7	3.63	7.44	17.20	3600
		8	4.20	7.31	20.18	3600
	7	6	3.36	5.92	19.20	3600
		7	3.64	8.30	24.81	3600
		8	4.49	9.96	25.38	3600
	8	6	3.70	7.04	26.05	3600
		7	4.51	11.13	26.58	3600
		8	10.64	14.06	27.40	3600

References

Akbar, M., & Irohara, T. (2018). Scheduling for sustainable manufacturing: A review. *Journal of Cleaner Production, 205*, 866–883.

Biel, K., & Glock, C. H. (2016). Systematic literature review of decision support models for energy-efficient production planning. *Computers & Industrial Engineering, 101*, 243–259.

Biel, K., Zhao, F., Sutherland, J. W., & Glock, C. H. (2018). Flow shop scheduling with grid-integrated onsite wind power using stochastic MILP. *International Journal of Production Research, 56*(5), 2076–2098.

Bozorgirad, M. A., & Logendran, R. (2013). Bi-criteria group scheduling in hybrid flowshops. *International Journal of Production Economics, 145*(2), 599–612.

Cheng, C. Y., Pourhejazy, P., Ying, K. C., & Liao, Y. H. (2021). New benchmark algorithms for no-wait flowshop group scheduling problem with sequence-dependent setup times. *Applied Soft Computing, 111*, Article 107705.

Cheng, J., Chu, F., Liu, M., Wu, P., & Xia, W. (2017). Bi-criteria single-machine batch scheduling with machine on/off switching under time-of-use tariffs. *Computers & Industrial Engineering, 112*, 721–734.

Costa, A., Cappadonna, F. V., & Fichera, S. (2020). Minimizing makespan in a Flow Shop Sequence Dependent Group Scheduling problem with blocking constraint. *Engineering Applications of Artificial Intelligence, 89*, Article 103413.

Cui, W., & Lu, B. (2021). Energy-aware operations management for flow shops under TOU electricity tariff. *Computers & Industrial Engineering, 151*, Article 106942.

Fang, K., Uhan, N. A., Zhao, F., & Sutherland, J. W. (2016). Scheduling on a single machine under time-of-use electricity tariffs. *Annals of Operations Research, 238*(1–2), 199–227.

Feng, H., Xi, L., Xiao, L., Xia, T., & Pan, E. (2018). Imperfect preventive maintenance optimization for flexible flowshop manufacturing cells considering sequence-dependent group scheduling. *Reliability Engineering & System Safety, 176*, 218–229.

Gahm, C., Denz, F., Dirr, M., & Tuma, A. (2016). Energy-efficient scheduling in manufacturing companies: A review and research framework. *European Journal of Operational Research, 248*(3), 744–757.

Giret, A., Trentesaux, D., & Prabhu, V. (2015). Sustainability in manufacturing operations scheduling: A state of the art review. *Journal of Manufacturing Systems, 37*, 126–140.

Ho, M. H., Hnaien, F., & Dugardin, F. (2020). Electricity cost minimisation for optimal makespan solution in flow shop scheduling under time-of-use tariffs. *International Journal of Production Research, 59*(4), 1041–1067.

European Commission. (2020). *JRC Photovoltaic Geographical Information System (PVGIS)*. https://re.jrc.ec.europa.eu/pvg_tools/en/.

Li, J., qing, Sang, yan, H., Han, Y. yan, Wang, C. gang, & Gao, K. zhou. (2018). Efficient multi-objective optimization algorithm for hybrid flow shop scheduling problems with setup energy consumptions. *Journal of Cleaner Production, 181*, 584–598.

Li, S. (1997). A hybrid two-stage flowshop with part family, batch production, major and minor set-ups. *European Journal of Operational Research, 102*(1), 142–156.

Liu, C.-H. (2015). Mathematical programming formulations for single-machine scheduling problems while considering renewable energy uncertainty. *International Journal of Production Research, 54*(4), 1122–1133.

Liu, C.-H. (2016). Discrete lot-sizing and scheduling problems considering renewable energy and CO2 emissions. *Production Engineering, 10*(6), 607–614.
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.
Mitrofanov, S. P. (1966). *Scientific principles of group technology: (Nauchnye Osnovy Gruppovoĭ Tekhnologii).* National Lending Library for Science and Technology.
Moon, J.-Y., & Park, J. (2014). Smart production scheduling with time-dependent and machine-dependent electricity cost by considering distributed energy resources and energy storage. *International Journal of Production Research, 52*(13), 3922–3939.
Mouzon, G., Yildirim, M. B., & Twomey, J. (2010). Operational methods for minimization of energy consumption of manufacturing equipment. *International Journal of Production Research, 45*(18–19), 4247–4271.
Nawaz, M., Enscore, E. E., & Ham, I. (1983). A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega, 11*(1), 91–95.
Neufeld, J. S., Gupta, J. N. D., & Buscher, U. (2016). A comprehensive review of flowshop group scheduling literature. *Computers & Operations Research, 70*, 56–74.
NREL transforming energy. (2020). *Distributed generation renewable energy estimate of costs | energy analysis | NREL* https://www.nrel.gov/analysis/tech-lcoe-re-cost-est.html.
Pan, Q. K., Gao, L., & Wang, L. (2020). An effective cooperative co-evolutionary algorithm for distributed flowshop group scheduling problems. *IEEE Transactions on Cybernetics*.
Pinedo, M. L. (2012). Scheduling: Theory, algorithms, and systems: Fourth edition. *Scheduling: Theory, Algorithms, and Systems: Fourth Edition*, 1–673 9781461423614.
Salmasi, N., Logendran, R., & Skandari, M. R. (2010). Total flow time minimization in a flowshop sequence-dependent group scheduling problem. *Computers & Operations Research, 37*(1), 199–212.
Schaller, J. E., Gupta, J. N. D., & Vakharia, A. J. (2000). Scheduling a flowline manufacturing cell with sequence dependent family setup times. *European Journal of Operational Research, 125*(2), 324–339.
Schulz, S., Neufeld, J. S., & Buscher, U. (2019). A multi-objective iterated local search algorithm for comprehensive energy-aware hybrid flow shop scheduling. *Journal of Cleaner Production, 224*, 421–434.
Shahvari, O. (2017). *Bi-criteria batching and scheduling in hybrid flow shops*. Oregon State University https://ir.library.oregonstate.edu/concern/graduate_thesis_or_dissertations/bk128g73p?locale=en.
Tang, D., Dai, M., Salido, M. A., & Giret, A. (2016). Energy-efficient dynamic scheduling for a flexible flow shop using an improved particle swarm optimization. *Computers in Industry, 81*, 82–95.
Wang, S., Zhu, Z., Fang, K., Chu, F., & Chu, C. (2017). Scheduling on a two-machine permutation flow shop under time-of-use electricity tariffs. *International Journal of Production Research, 56*(9), 3173–3187.
Wang, X., Ding, H., Qiu, M., & Dong, J. (2011). A low-carbon production scheduling system considering renewable energy. In *Proceedings of 2011 IEEE international conference on service operations, logistics and informatics* (pp. 101–106).
Willey, P. (1975). The introduction of group technology. *Production Engineer*. Institution of Engineering and Technology (IET) (Vol. 54, Issue 12).
Wu, X., & Sun, Y. (2018). A green scheduling algorithm for flexible job shop with energy-saving measures. *Journal of Cleaner Production, 172*, 3249–3264.
Zhai, Y., Biel, K., Zhao, F., & Sutherland, J. W. (2017). Dynamic scheduling of a flow shop with on-site wind generation for energy cost reduction under real time electricity pricing. *CIRP Annals, 66*(1), 41–44.
Zhang, H. (2016). *Flow shop scheduling for energy efficient manufacturing*. Purdue University https://docs.lib.purdue.edu/open_access_dissertations/891.
Zhang, H., Cai, J., Fang, K., Zhao, F., & Sutherland, J. W. (2017). Operational optimization of a grid-connected factory with onsite photovoltaic and battery storage systems. *Applied Energy, 205*, 1538–1547.
Zhang, H., Zhao, F., Fang, K., & Sutherland, J. W. (2014). Energy-conscious flow shop scheduling under time-of-use electricity tariffs. *CIRP Annals, 63*(1), 37–40.