Scientia Iranica E (2018) 25(1), 339–358

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
http://scientiairanica.sharif.edu

A flexible cell scheduling problem with automated guided vehicles and robots under energy-conscious policy

M. Hemmati Far[a], H. Haleh[b,*], and A. Saghaei[a]

a. *Department of Industrial Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.*
b. *Department of Industrial & Mechanical Engineering, Islamic Azad University, Qazvin Branch, Qazvin, Iran.*

Received 29 October 2015; received in revised form 4 July 2016; accepted 8 November 2016

KEYWORDS
Cell-scheduling;
Automated Guided Vehicles (AGV);
Robots;
Energy-conscious policy;
Ant Colony Optimization (ACO);
Genetic Algorithm (GA);
Taguchi experimental design method.

Abstract. A flexible Cell Scheduling Problem (CSP) under Time-Of-Use (TOU) electricity tariffs is developed in this study. To apply a kind of energy-conscious policy, overconsumption cost of on-peak period electricity, limitations on total energy consumption by all facilities, setup time available on each cell, part defect (pert) percentage, and the total number of Automated Guided Vehicles (AGV) were considered. Additionally, an Ant Colony Optimization (ACO) algorithm was employed to find a near-optimum solution to the proposed Mixed Integer Linear Programming (MILP) model with the objective of minimizing the total cost of CSP model. Since no benchmark is available in the literature, a lower bound was implemented as well to validate the result achieved. Moreover, to improve the quality of the results obtained by meta-heuristic algorithms, two hybrid algorithms (HGA and HACO) were proposed to solve the model. For parameter tuning of algorithms, Taguchi experimental design method was applied. Then, numerical examples were presented to prove the application of the proposed methodology. Our results were compared with the lower bound, confirming consequently that HACO is capable of finding better and near-optimum solutions.

1. Introduction

Cellular Manufacturing System (CMS) is a production system in which parts are collected into dedicated manufacturing cells, consistent with a number of relationships in their design and manufacturing features. The key benefits of CMS described in the literature are: less production cost, less material handling cost, decline in throughput time and Work-In-Process (WIP), and also simple production control [1]. With the improvement of manufacturing technology and coping with the challenges of customer's demands from mass production to medium product diversity and medium or small production volumes with shorter product life cycles, CMSs received more courtesy from investigators and manufacturers [2-6]. Cell Part Scheduling problem (CPS) involves the allocation of manufacturing resources over time to perform the processes on a collection of parts in cells, which is a decision-making process that plays a significant role in CMS [7]. Based on the number of cells, the CPS problem can be divided into Single-Cell Part Scheduling (S-CPS) and Multi-

*. *Corresponding author. Tel.: +98 21 88534460; Fax : +98 21 88534461*
E-mail addresses: m.hemmatifar@gmail.com (M. Hemmati Far); hhaleh@gut.ac.ir (H. Haleh); a.saghaei@srbiau.ac.ir (A. Saghaei)

doi: 10.24200/sci.2017.4395

Cell Part Scheduling (M-CPS). When dealing with S-CPS, the problems are comparable to classical flow-shop or job-shop scheduling problems, which are NP-hard problems. Therefore, some heuristics or meta-heuristic algorithms are proposed [8-10]. A Flexible Manufacturing System (FMS) can be defined as an integrated arrangement of Numerical Control (NC) machine tools, some supplementary production equipment, and a material handling system designed to simultaneously manufacture low to medium volumes of a wide variety of high-quality goods at low cost [11]. FMS is generally classified into four different categories: Single Flexible Machines (SFMs), Flexible Manufacturing Cells (FMCs), Multi-Machine FMSs (MMFMSs), and Multi-Cell FMSs (MCFMSs) [12].

Automated Guided Vehicles (AGVs) are among various innovative material handling methods that are finding increasing applications today. They can be interfaced to different other production and storage equipment and controlled through an intelligent computer control system. Both the scheduling of operations on machines as well as the scheduling of AGVs are crucial elements, contributing to the efficiency of the overall FMS [13]. An AGV material handling system, which is typically integrated with a block layout, is frequently defined by the material handling network. This network specifies the configuration and direction of the network as well as the number and location of P/D stations [14,15]. On the world-class level, the field of application of mobile robots is not constrained to the industry; it is considerably wider, also reaching the areas of logistics (distribution and storage), oceanographic and underwater exploration, planetary exploration, and military applications. At this time in the industry, and specifically in the existing industrial projects of mobile robotics, the core objective applications are in manufacturing (factories, cells, and FMSs), storage, and services in the logistics of supply chain [16].

The growth in price and demand for petroleum and other fossil fuels, together with the decrease in the reserves of energy commodities and the increasing concern over global warming, has resulted in greater efforts toward the minimization of energy consumption. In the USA, the manufacturing sector consumes about one-third of the energy usage and produces about 28% of greenhouse gas (GHG) emissions. To produce one kilowatt-hour of electricity, two pounds of carbon dioxide is released into the atmosphere, therefore contributing to global warming [17].

Although a substantial number of research papers are available in the literature, a brief review of those on the Cell Scheduling Problem (CSP) is presented in the next section.

2. Literature review

Generally, studies of scheduling problem can be categorized into six main categories: FMS, CSP, FMC, flow shop, job shop, and assembly line, respectively.

2.1. FMS studies

Scheduling problem for FMS has been investigated by some academics: Ulusoy et al. [18], Ali et al. [19], Pach et al. [20], and Leitao & Restivo [21]. After that, Abazari et al. [22] proposed a linear mathematical programming model with both continuous and zero-one variables for job selection and operation-allocation problems in an FMS to maximize profitability and utilization of system. The proposed model assigns operations to different machines considering capacity of machines, batch-sizes, processing time of operations, machine costs, tool requirements, and capacity of tool magazine. They used a GA to solve the formulated problem.

Similarly, Pach et al. [23] optimized FMS by a new general hybrid control architecture called ORCA (dynamic Architecture for an Optimized and Reactive Control). Balaji and Porselvi [24] used an artificial immune system and Simulated Annealing (SA) algorithms for scheduling batches of parts based on job availability model in a multi-cell FMS. A mathematical model for the investigation problem is used with the objective function of minimizing the make-span. In addition, Erdin and Atmaca [25] proposed the implementation of the overall design of a FMS. They used an application of the strategy of an FMS to determine essential number, utilization and sequence of workstations and plant layout for the assumed manufactured quantity of diverse parts, processing sequence, and times. He et al. [26] investigated the problems of part input sequencing and scheduling in FMS in a Mass Customization/Mass Personalization (MC/MP) environment. They considered both robot and machine scheduling rules using a state-dependent part input sequencing algorithm.

2.2. CSP studies

Considering different aspects including machine grouping control, Virtual Manufacturing Cells (VMCs), flow-line manufacturing cell, group layout design model, etc., CSP has been studied by some researchers: Solimanpur et al. [27], Logendran et al. [28], Venkataramanaiah [29], Tavakkoli-Moghaddam et al. [30], Lin et al. [31], Shirazi et al. [32], Kesen et al. [33], Lin et al. 2011 [34], Kia et al. [35], Batur et al. [36], Izui et al. [37], Boutsinas [38], and Fazlollahtabar and Jalali Naini [39]. Later, Zeng et al. [40] studied the problem of parts scheduling in multi-job shop cells by considering exceptional parts that need to visit machines in different cells and be transferred by robot with the objective of minimizing the make-

span. Forghani Mohammadi [41] presented a unified methodology to consecutively solve the cell formation and layout problems. Also, in order to measure the material handling cost more accurately, the real situation of the machines within the cells was used (instead of the center-to-center distances between the cells). Correspondingly, Zhang et al. [42] developed a time-indexed integer programming formulation to identify manufacturing schedules that minimize electricity cost and the carbon footprint under TOU tariffs without compromising production throughput. Their results suggested that shifting electricity usage, while reducing electricity cost, may increase CO_2 emissions in the regions where the grid base load is met with electricity from coal-fired power plants.

Li et al. [43] considered the Flow-line Manufacturing CSP (FMCSP) with Sequence-Dependent Family Setup Times (SDFSTs) for total tardiness and mean total flow time minimization. Based on the mathematical model of this problem, a Hybrid Harmony Search (HHS) was proposed. Experimental results from 900 problem instances showed that HHS performs relatively better than these meta-heuristics in finding schedules to minimize the multi-objective FMCSP with SDFSTs. Recently, Zohrevand et al. [44] have developed a bi-objective stochastic model. The first objective function of the developed model seeks to minimize the total cost, and the second objective function maximizes labor utilization of the cellular manufacturing system. Also, Majumder and Laha [45] investigated the problem of 2-machine robotic cell scheduling of one-unit cycle with sequence-dependent setup times and different loading/unloading times of the parts. They implemented a discrete cuckoo search algorithm offered to determine the sequence of robot moves along with the sequence of parts so that the cycle time is minimized.

2.3. FMC category

Gultekin et al. [46] considered a bi-criteria model which decides on the allocation of the operations to the machines, the processing times of the operations on the machines, and the robot move sequence that jointly minimize the cycle time and the total manufacturing cost. Tüysüz and Kahraman [47] presented a method for modeling and exploration of time-critical, dynamic and complex systems using stochastic PN together with fuzzy sets. Recently, Naderi and Azab [48] have studied scheduling of a FMC with parallel processing capability. They formulated a mixed integer linear programming model and used five meta-heuristics algorithms to solve it. Also, Yang et al. [49] implemented an improved discrete Particle Swarm Optimization (PSO) with genetic operator and random-heuristic initialization method to minimize the make-span of Flexible robotic manufacturing CSP with multiple robots.

2.4. Flow shop studies

In the flow shop issue, there exist some topics involving group scheduling problem [50-52], flexible flow shop [53], flow shop scheduling [54-57] energy-based flow shops, etc. With respect to the energy application in the flow shop, Fang et al. [58] presented a general multi-objective mixed integer linear programming formulation for optimizing the operating schedule of a flow shop that considered both productivity (i.e., make-span) and energy (i.e., peak load and carbon footprint) related criteria. Dai et al. [59] proposed an energy-efficient model for Flexible Flow Shop scheduling (FFS). First, a mathematical model for a FFS problem, which is based on an energy-efficient mechanism, is described to solve multi-objective optimization. Since FFS is well known as a NP-hard problem, an improved, genetic-simulated annealing algorithm is adopted to make a critical trade-off between the make-span and the total energy consumption to implement a feasible scheduling. Finally, a case study of a production scheduling problem for a metal working workshop in a plant is simulated. The experimental results show that the relationship between the make-span and the energy consumption may be apparently conflicting. In addition, an energy-saving decision is performed in a feasible scheduling.

2.5. Job shop researches

In the job shop problem, there exist two subjects including job shop scheduling [60] and flexible job shop scheduling [61-63].

2.6. Assembly studies

In this topic, the assembly line balancing problem was investigated by Hamta et al. [64] and Khalili et al. [65].

As seen, to the best of author's knowledge, there exists no research work which has considered setup time, part defect (pert) percentage, and the total number of AGV simultaneously in the flexible cell scheduling problem under Time-Of-Use (TOU) electricity tariffs. This research was motivated by Abazari et al. [22] and Zhang et al. [42] studies, in which a flexible cell scheduling problem under Time-Of-Use (TOU) electricity tariffs is developed. Moreover, to bring applicability closer to their model of real energy-conscious problems, additional limitations, such as total available consumed power, setup time, part defect (pert) percentage, and the total number of AGV are considered. Additionally, two meta-heuristic algorithms (GA and ACO) are utilized to find a near-optimum solution to the proposed Mixed Integer Linear Programming (MILP) model with the objective of minimizing the total cost of CSP model. To increase the quality of the outcomes obtained by meta-heuristic algorithms, two hybrid algorithms (HGA and HACO) are proposed to solve the model. The hybrid outcomes

are compared with the lower bound for more validation. In addition, Taguchi experimental design method is applied to parameter tuning of the algorithms. In short, the highlights of the differences of this research from the above-mentioned studies are as follows:

- Considering time-of-use electricity tariffs and limitation on total power consumption to make an energy-conscious model;
- Adding further constraints to the model, such as setup time available, part defect (pert) percentage, and the total number of AGV, to make the model more applicable and extend the model of Cell Scheduling Problem (CSP);
- Considering an Ant Colony Optimization (ACO) and a Genetic Algorithm (GA) to better solve the new model and compare the results;
- Developing two hybrid algorithms (HACO and HGA) to improve quality of the results;
- Using Taguchi experimental design method for parameters tuning of algorithms;
- Comparing the solutions of HACO with a lower bound for more justification.

The structure of the rest of the paper is organized as follows. In Section 3, the problem is defined and the assumptions are made. In Section 4, the problem is mathematically formulated into a Mixed Integer Linear Programming (MILP) model. An ACO and a GA are proposed to solve the problem in Section 5, and also a Taguchi experimental design method is utilized for parameter tuning of algorithms. In order to demonstrate the application of the proposed approach, numerical examples are solved in Section 6. Finally, conclusions and future research topics are provided in Section 7.

3. The problem and assumptions

3.1. The problem

CMS has a structure that can present application of modern manufacturing knowledge, e.g. just-in-time manufacturing, FMSs, and computer-integrated manufacturing, where a number of diverse jobs come with various processing necessities. A job contains one or more operations and each of them can be executed by one or more cells. It is assumed that the particularities related to the production requirements of the job, number of operations for each job and their machining time, and number of tool slots required by each operation of each job on every cell are identified in advance. Also, critical and optional types of operations are allied with each job. Vital operations of a job mean that this operation can be applied only to a specific cell using a deterministic number of tool slots, whereas optional operations imply that they can be carried out on a number of cells with the same or varying machines, processing time, and tool slots. In this problem, the flexibility lies in the selection of a cell for processing the optional operations of the jobs.

It must be highlighted that the European Union legalized guidelines on energy like Directive of the European Parliament and of the Council 2006/32/WE from 5th April, 2006 endue efficiency and energy services and repeal Council Directive 93/76/EEC. Based on the universal developments and latest progress in green technologies, there is a requirement to take into consideration the energy consumption and CO_2 emissions in a company with the usually applied objectives (minimum travel time, maximum throughput, and minimum cost). Therefore, in the present study, the energy efficiency model for flexible Cell Scheduling Problem (CSP) is presented and evaluated. This research is concerned with a CSP in a FMC under Time-Of-Use (TOU) electricity tariffs in which not only the time available for each cell, but also the maximum setup time available for each cell are limited; it is also concerned with two AGVS for material handling system. Furthermore, there are bounds on the total energy consumption and the pert percentage that it is fixed and independent of any type of job operations. The objective is to find the near-optimum solution to job selection and loading of a Mixed Integer Linear Programming (MILP) model so that the total cost, including under-utilized cost, over-utilized cost, setup times cost, pert percentage cost, operation process cost, and energy consumption cost, is minimized while the constraints are fulfilled.

3.2. Assumptions

In order to minimize the complexities of evaluating the problem for an applied FMC as shown in Figure 1, the following assumptions are used for the mathematical formulation of our model:

(a) All considered times for wholly jobs are deterministic;

(b) In each cell, duty of robots is to load or unload like AGVS;

(c) The times of loading or unloading between robots with machines or AGVS are similar;

(d) Every AGV can have only one batch of each job in order to process the receipt of material storage;

(e) Transportation of AGVs is without delay;

(f) During AGVs travel, stopping is not feasible;

(g) The travel of AGVs is full or empty depending on the existence or non-existence of job;

(h) Movement of AGVs is based on the shortest route;

(i) Travel time of AGVs depends on sequence of jobs;

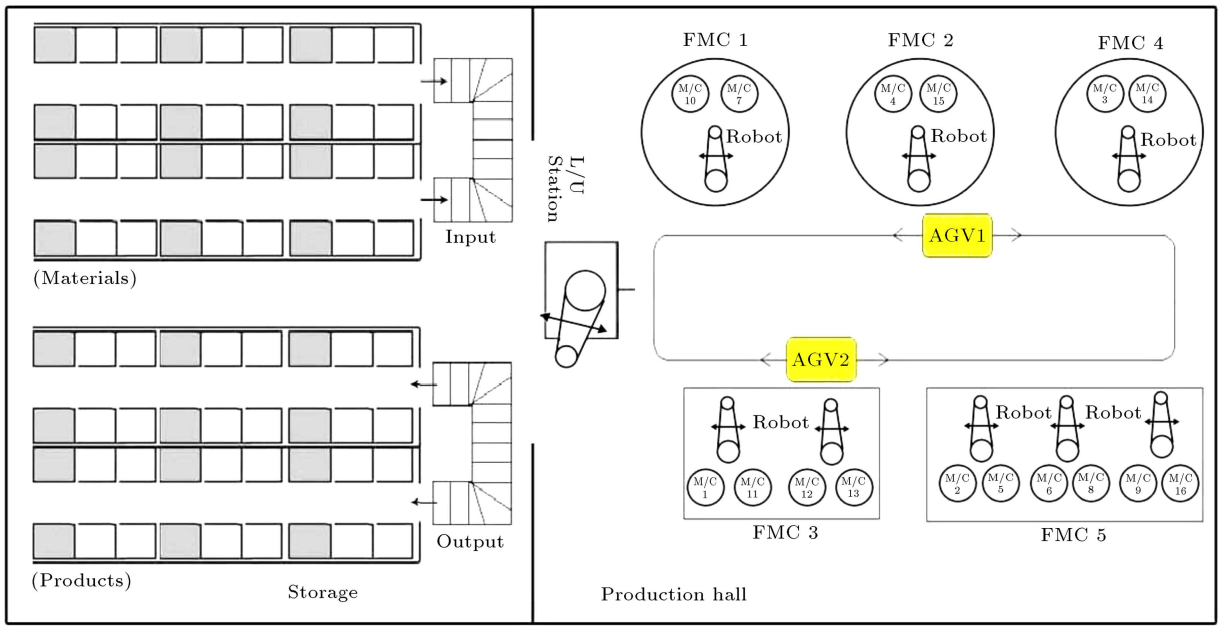

Figure 1. Layout of FMC.

(j) In each time, one empty AGV is available;

(k) All AGVS start the movement of loading or unloading station at time zero;

(l) All the jobs, machines, tools, AGVS, robots, and cells are available at time zero;

(m) Each operation process has to be performed without preemption on exactly one resource of a specific resource type;

(n) Sharing of tool slots, machines, AGVS, and robots is not considered;

(o) Each machine can process maximum one operation at a time;

(p) Adequate rapidly is available for loading or unloading;

(q) Over-loading and under-loading of cells are acceptable;

(r) The setup times and pert percentage available for each cell are limited;

(s) Each cell has a fixed pert percentage for processing any operation (independent of jobs);

(t) The operations of a job must be completed on the same cell once a cell is selected;

(u) One robot in each cell for loading or unloading is considered;

(v) Energy consumption as a constraint is considered.

4. Mathematical model

Before presenting the mathematical formulation of the problem at hand, the notations are first introduced.

The parameters and variables of the model are defined as follows:

4.1. Parameters

C	Number of cells
M	Number of machines
L	Number of tool slots
G	Number of AGVS
O	Number of operation for each jobs
J	Number of jobs
O_j	Number of operations of job j
c	Index of cell; $1 \leq c \leq C$
m	Index of machine; $1 \leq m \leq M$
l	Index of tool slots; $1 \leq l \leq L$
g	Index of AGVS; $1 \leq g \leq G$
o	Index of operation; $1 \leq o \leq O_j$
j	Index of job; $1 \leq j \leq J$
X_o	Horizontal location coordinate of storage
Y_o	Vertical location coordinate of storage
X_c	Horizontal location coordinate of cell c
Y_c	Vertical location coordinate of cell c
T_c	Available time on cell c
V_{AGV}	Speed of full AGV
V'_{AGV}	Speed of empty AGV
C_c	Idleness unit cost of cell c
C'_c	Over time unit cost of cell c
$C_{ST(joml)}$	Setup time unit cost for operation o of job j on machine m by toll slot l

α_{joml}	Pert percentage for operation o of job j on machine m by toll slot l
$C_{\alpha(joml)}$	Pert unit cost for operation o of job j on machine m by toll slot l
$C_{op(joml)}$	Process unit cost for operation o of job j on machine m by toll slot l
$U_{\alpha(c)}$	Upper limit for allowable pert percentage capacity of cell c
b_j	Batch size of job j
t_{joml}	Processing time required for operation o of job j on machine m by toll slot l
ST_c	Setup time capacity of cell c
ST_{joml}	Setup time required for operation o of job j on machine m by toll slot l
P_M	Quantity of electric energy consumption by one machine (kWs)
P_b	Quantity of electric energy consumption by one robot or AGV (kWs)
P	Total allowable electric energy consumption of production system (kWs)
tl_j	Time of loading or unloading for job j by robot
$C_{p(\text{medium})}$	Unit cost (tariff) of kW power used by a facility in mid-peak hours
$C_{p(\text{high})}$	Difference unit cost of kW power used by a facility in mid-peak hours with on-peak hours
$B(j,o)$	Set of cells that can perform operation o of job j

4.2. Variables

TPC	Total electric energy consumption of production system (kWs)
TPC_1	Quantity of electric energy consumption for all machines (kWs)
TPC_2	Quantity of electric energy consumption for all robots and AGVS (kWs)
TPC_3	Total of electric energy consumption on-peak hours of electric energy consumption
$TC(I)$	Total cost of under-utilized of all facilities
$TC(O)$	Total cost of over-utilized of all facilities
$TC(\text{other})$	Other costs including setup times cost, pert cost, and process cost
$TC(P)$	Total cost of electric energy consumption of all facilities
$TC(\text{medium})$	Total cost of power consumption in mid-peak hours for all facilities
$TC(\text{high})$	Total cost of power consumption in on-peak hours for all facilities
U_c	Under-utilized time on cell c (a decision variable)
O_c	Over-utilized time on cell c (a decision variable)
O_p	Over-utilized electric energy consumption of all the facilities (a decision variable)
U_p	Under-utilized electric energy consumption of all the facilities (a decision variable)

$$X_{jogcml} = \begin{cases} 1 & \text{if operation } o \text{ of job } j \text{ is assigned} \\ & \text{on machine } m \text{ in cell } c \text{ by tool slot} \\ & l \text{ and carried AGV}_g \\ 0 & \text{otherwise} \end{cases}$$

$$Y_j = \begin{cases} 1 & \text{if job } j \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

$$AGV_g = \begin{cases} 1 & \text{if AGV } g \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

Based on the above definitions, the mathematical model of the problem is derived in the next subsections.

4.3. The objective function cost

The total cost of objective function includes under-utilized cost, over-utilized cost, setup times cost, pert cost, process cost, and electric energy consumption cost. With respect to the total available time of all cells which is limited (see Eq. (9)), we have under-utilized time cost and over-utilized time cost as follows:

$$TC(I) = \sum_{c=1}^{c} (U_c.C_c), \tag{1}$$

$$TC(O) = \sum_{c=1}^{c} (O_c.O'_c). \tag{2}$$

Another cost function in the model, involving setup times cost, pert cost, and process cost for all of cells, is as follows:

$$TC(\text{other}) = \sum_{c=1}^{C} \sum_{l=1}^{L} \sum_{o=1}^{O_j} \sum_{j=1}^{J} \sum_{m=1}^{M} C_{ST(joml)}.X_{jogcml} .ST_{joml} + \sum_{c=1}^{C} \sum_{l=1}^{L} \sum_{o=1}^{O_j} \sum_{j=1}^{J} \sum_{m=1}^{M} C_{\alpha(joml)}$$

$$.X_{jogcml}.b_j.\alpha_{joml} + \sum_{c=1}^{C}\sum_{l=1}^{L}\sum_{o=1}^{O_j}\sum_{j=1}^{J}\sum_{m=1}^{M}$$

$$C_{OP(joml)}.t_{joml}.X_{jogcml}.b_j. \quad (3)$$

In addition, the total cost of consumed power, involving mid-peak hours and on-peak hours for all facilities, is as follows:

$$TC(\text{medium}) + TC(\text{high}) = \left(TPC.C_{p(\text{medium})}\right) + \left(TPC_3.C'_{p(\text{high})}\right), \quad (4)$$

where:

$$TPC = TPC_1 + TPC_2. \quad (5)$$

The quantity of consumed power for all of machines (TPC_1), robots, and AGVS (TPC_2) is, respectively, as follows:

$$TPC_1 = P_M.\left[\sum_{c=1}^{C}\sum_{j=1}^{J}\sum_{o=1}^{O_j}\sum_{l=1}^{L}\sum_{m=1}^{M}\left(b_j.t_{joml}.X_{jogcml} + ST_{joml}.X_{jogcml}\right)\right]. \quad (6)$$

$$TPC_2 = P_b.\left[2.\left(\sum_{j=1}^{J} tl_j.Y_j\right) + 2.\left(\sum_{j=1}^{J} b_j.tl_j.Y_j\right) + 2.\left(\sum_{g=1}^{G}\sum_{j=1}^{J}\text{AGV}_g.\frac{|X_o - X_c| + |Y_o - Y_c|}{V_{\text{AGV}}}.Y_j + \sum_{g=1}^{G}\sum_{j=1}^{J}\text{AGV}_g.\frac{|X_o - X_c| + |Y_o - Y_c|}{V'_{\text{AGV}}}.Y_j\right)\right]. \quad (7)$$

Furthermore, total consumed power in on-peak hours for all equipment is:

$$TPC_3 = (P_M + P_b).\sum_{c=1}^{C} O_c. \quad (8)$$

4.4. The constraints

As mentioned previously, under-utilized time of a cell is treated as the unused capacity of that machine, whereas over-utilized time is considered as the overload on a cell [22]. We accepted under-utilized time and over-utilized time of cells, that is:

$$\sum_{j=1}^{J}\sum_{o=1}^{O_j}\sum_{l=1}^{L}\sum_{m=1}^{M}\left(b_j.t_{joml}.X_{jogcml} + ST_{joml}.X_{jogcml}\right) + U_c - O_c = T_c, \quad c = 1, 2, ..., C. \quad (9)$$

Since the required setup times for the operations of the jobs to be completed on a cell must always be less than or equal to the setup times available on that cell, we have:

$$\sum_{j=1}^{J}\sum_{o=1}^{O_j}\sum_{l=1}^{L}\sum_{m=1}^{M} ST_{joml}.X_{jogcml} \le ST_c, \quad c = 1, 2, ..., C. \quad (10)$$

In addition, the total pert percentage occurring for the operations of the jobs to be completed on a cell should be always less than or equal to the allowable pert percentage limit on that cell, that is:

$$\sum_{j=1}^{J}\sum_{o=1}^{O_j}\sum_{l=1}^{L}\sum_{m=1}^{M} X_{jogcml}.b_j.\alpha_{joml} \le U_{\alpha(c)}.\sum_{j=1}^{J} b_j, \quad c = 1, 2, ..., C. \quad (11)$$

Furthermore, the total number of AGV in production system is limited, that is:

$$1 \le \sum_{g=1}^{G}\text{AGV}_g \le 2. \quad (12)$$

Also, the total allowable consumed power by all facilities is restricted to (P), that is:

$$P_M.\left[\sum_{c=1}^{C}\sum_{j=1}^{J}\sum_{o=1}^{O_j}\sum_{l=1}^{L}\sum_{m=1}^{M}\left(b_j.t_{joml}.X_{jogcml} + ST_{joml}.X_{jogcml}\right)\right] + P_b.\left[2.\left(\sum_{j=1}^{J} tl_j.Y_j\right) + 2.\left(\sum_{j=1}^{J} b_j.tl_j.Y_j\right) + 2.\left(\sum_{g=1}^{G}\sum_{j=1}^{J}\text{AGV}_g.\frac{|X_o - X_c| + |Y_o - Y_c|}{V_{\text{AGV}}}.Y_j + \sum_{g=1}^{G}\sum_{j=1}^{J}\text{AGV}_g.\frac{|X_o - X_c| + |Y_o - Y_c|}{V'_{AGV}}.Y_j\right)\right] + U_p - O_p = P. \quad (13)$$

Additionally, once a job is selected, each operation of that job should be completed by one and only one cell. Correspondingly, if a job is not selected, no cell is assigned to perform any operation of that job; so, we have:

$$\sum_{c\in B(j,o)} X_{jogcml} = Y_j, \quad Y_j = 0,1, \quad j = 1,2,...,J,$$

$$o = 1,2,...,O_j, \quad g = 1,2, \quad c = 1,2,...,C,$$

$$m = 1,2,...,M, \quad l = 1,2,...,L, \tag{14}$$

where decision variables are binary as:

$$X_{jogcml} = 0,1, \quad Y_j = 0,1. \tag{15}$$

In addition, all decision variables should be non-negative as:

$$U_c, O_c, \geq 0, \quad U_p, O_p \geq 0. \tag{16}$$

We considered objective function as the total cost minimization of a flexible CSP under TOU electricity tariffs so that all the constraints are fulfilled. In the next section, a hybrid solution algorithm is presented to efficiently solve the problem.

5. The hybrid solution algorithms

Abazari et al. [22] mentioned the number of possible solutions some of which may not be feasible with respect to the constraints of system such as capacity of machines/cells and capacity of tool magazine. Computing all these solutions to determine the optimum one is computationally intractable for medium- to large-sized problems. In addition, it is the fact that MLP of a FMS is recognized for its complexity [66]. Moreover, the MLP related to automated manufacturing system belongs to the classification of NP-hard problems where the computational solution times are non-polynomial in the size of the problem [67-72]. Due to this fact, we need to employ a meta-heuristic search algorithm to solve it. Therefore, two meta-heuristic algorithms are employed as well to enable the validation of the results obtained. In the next two subsections, brief descriptions are first given for GA and ACO.

Recent investigations in [73-76] have shown that hybrid meta-heuristics work better than individual meta-heuristics in solving mixed integer or nonlinear models. Commonly, hybridization refers to the mixture of two search algorithms to solve a given problem [75]. There are some methods to utilize hybrid meta-heuristics, one of which is combining the traditional GA with any of the other meta-heuristic algorithms. Consequently, this research considered a hybrid algorithm based on a GA and an Ant Colony Optimization (ACO), named HACO, in order to solve the Mixed Integer Linear Programming (MILP) problem. In the proposed method, to make a hybrid algorithm (HGA), ACO is used to produce the best initial solutions. Accordingly, the initial inputs for the hybrid GA come from the best outputs ($U_C, O_C, U_P, O_P, Y_j, X_{jogcml}, \text{AGV}_g$) of the ACO. Then, The HGA runs until a termination condition (i.e., on the maximum number of iterations) is met.

5.1. Genetic Algorithm (GA)

Genetic algorithms are stochastic search methods based on the mechanism of natural selection and natural genetics. GA varies from conventional search methods in a sense that it starts with an initial set of random solutions, called population. Each individual in the population is called a chromosome, representing a solution to the problem at hand. The chromosomes evolve through successive iterations, called generations. During each generation, the chromosomes are evaluated through some measures of fitness. To produce the next generation, new chromosomes, called offspring, are shaped by either crossover or mutation operators. A new generation is formed according to the fitness values of the chromosomes. After several generations, the algorithm converges to the best chromosome [77].

The core parameters of a GA are population size, N_{GA}, crossover probability, P_c, and mutation probability, P_m. Also, the steps taken in the proposed real-coded GA algorithm are:

1. Set parameters P_c, P_m, and N_{GA};
2. Initialize the population randomly;
3. Evaluate the objective function (total cost) of all chromosomes;
4. Select individuals for mating pool;
5. Apply the crossover operation to each pair of chromosomes with probability P_c;
6. Apply the mutation operation to each chromosome with probability P_m;
7. Replace the current population by the resulting mating pool;
8. Evaluate the objective function;
9. If stopping criterion is met, stop. Otherwise, go to Step 5.

Figure 2 shows the flowchart of the GA algorithm [78].

5.2. The Ant Colony Optimization (ACO)

Ant Colony Optimization (ACO) is a meta-heuristic algorithm based on Swarm Intelligence (SI) suggested by Dorigo in the 1990s (e.g., [79-81]). It is one of the most advanced methods for approximate optimization that has been used to deal with many problems in real-world environments. In what follows, we briefly review the basis of ACO employed to find a near-optimum solution.

The notion behind ACO is based on the "natural" algorithm used by real ants to generate a near-optimal path between their nest and the food source, as shown

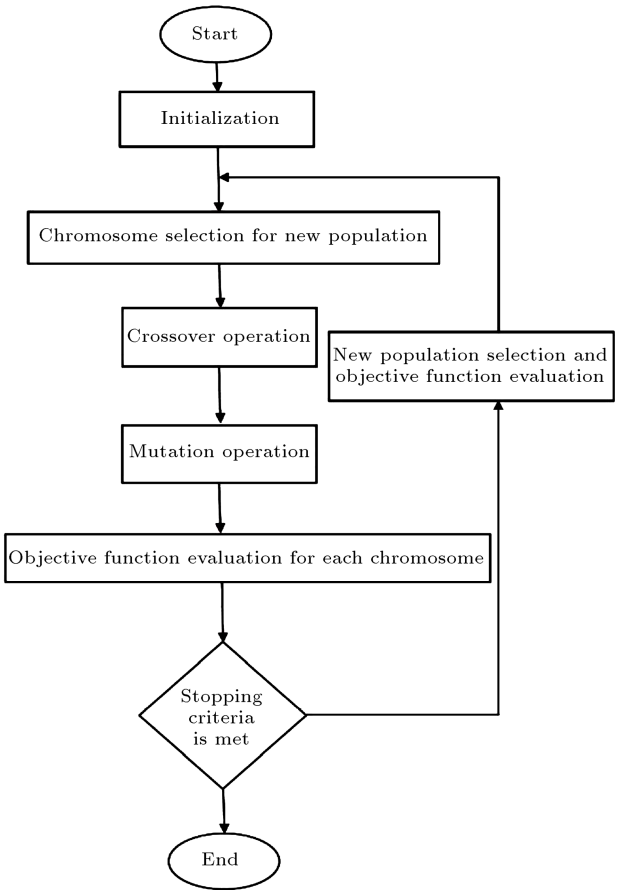

Figure 2. The flowchart of the proposed GA [78].

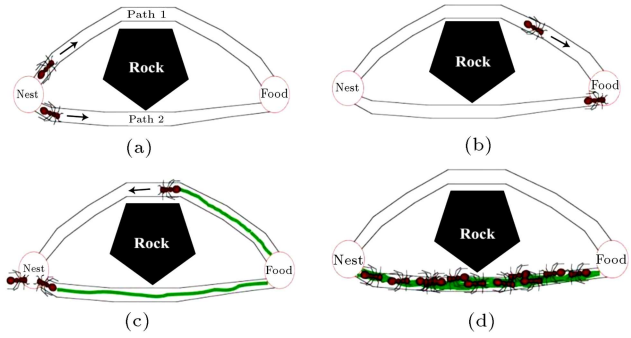

Figure 3. Basic Ant Colony Optimization behavior at different time stamps. The green areas represent the amount of pheromones on each path.

in Figure 3. During their seeking food process, ants deposit chemical substances, called pheromones, on their way back to their nest. Other ants sense the pheromone and get highly interested towards the marked paths; the more pheromone that is released on a path, the more attractive that path becomes. The pheromone evaporates and vanishes over time. Evaporation removes the pheromone on longer paths (and also on less interesting paths). Shorter paths are refreshed more rapidly, therefore having the chance of being more frequently explored. Naturally, ants will join towards

Initialize (set parameters)
repeat
Each ant is located on the primary node
repeat
Each ant employs a state move rule to increase the solution
Apply the pheromone local update rule
until all the ants have made a complete solution
Apply a local search method
Apply the pheromone global update rule
until the stop conditions is fulfilled

Figure 4. The procedure involved in the ACO algorithm.

the most efficient path due to the fact that it gets the strongest density of pheromone. The concept of the ACO algorithm is to mimic this performance. This simulation is achieved by creating a pheromone matrix, $n \times m$, employed by two key operations: the pheromone quantity tuning (also identified as pheromone deposit and pheromone evaporation rules) and a probabilistic rule that selects an endpoint based on the pheromone quantity (the state transition rule).

The procedure of the ACO algorithm is displayed in Figure 4. In this algorithm, m ants are employed in each cycle to make a full solution. To complete this task, the solution is obtained in steps. To define each step, two rules are used as follows:

$$s = \begin{cases} \arg \max\limits_{u \in N^*} \{[\tau(i,j)].[\eta(i,j)]^\beta\} \\ \qquad \text{if} q \leq q_0 \text{ (exploration)} \\ S \text{ otherwise (exploitation)} \end{cases} \tag{17}$$

$$p_{ij}^r(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha.[\eta_{ij}]^\beta}{\sum\limits_{j \in N^*}[\tau_{ij}(t)]^\alpha.[\eta_{ij}]^\beta} \text{ if } j \in N^*, \\ 0 \quad \text{otherwise} \end{cases} \tag{18}$$

where:

- i and j are two nodes on the graph that denote the search space;
- s is an arc that connects i and j;
- S is a path chosen according to the probability given in Eq. (18);
- q is a uniform random value between 0 and 1;
- $0 \leq q_0 \leq 1$ is a parameter chosen during the implementation of the algorithm;
- $0 \leq \alpha \leq 1, 0 \leq \beta \leq 1$ are two parameters that determine the relative influence of the pheromone path and heuristic information;

- N^* is the set of feasible nodes that can be visited by an ant;
- r denotes the index of the ant;
- p_{ij}^r denotes the probability of ant r in node i to choose node j;
- $\tau(i,j)$ is the pheromone path value between nodes i and j;
- $\eta(i,j)$ is a heuristic value used as the visibility from nodes i to j.

Also, two updated rules are used: the first is the evaporation of the existing pheromone; the second is the quantity of the added pheromone on the path. These rules are presented in Eqs. (19) and (20):

$$\tau_{ij}(t+1) = \rho.\tau_{ij}(t) + \sum_{r=1}^{m} \Delta_{ij}^{r}(t), \tag{19}$$

$$\Delta_{ij}^{r}(t) = \begin{cases} 1/L_r & \text{if ant } r \text{ goes from node } i \text{ to node } j \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

where:

- L_r shows how much the pheromone path should increase;
- $0 \leq \rho \leq 1$ is the evaporation parameter.

5.3. Taguchi experimental design method

A standard experimental policy mode was mainly established in 1920 by Ronald Fischer to expand the efficiency of farming production [82]. Genichi Taguchi advanced a different experimental design technique to increase the efficiency of application and appraisal of experiments. Its method is further appropriate for appraising production processes because the necessary number of experiments is reduced meaningfully. The design of experiment by the Taguchi method affords a simple, effective, and organized methodology to define the best settings [83,84]. Experimental results are changed to a Signal/Noise (S/N) ratio, which means a ratio of an average standard deviation (Taguchi et al. [85]). This ratio can be designed in three diverse methods: 'small value is good', 'great value is good', and 'nominal value is good'. A superior S/N ratio shows a better test effect. Thus, in experiments, a level of the factor which has the maximum ratio denotes a better presentation. The ratio lets control mean and variance, while an analysis of variance (ANOVA) is accomplished at the similar time. In this way, the effects of factors can be exposed statistically. S/N ratio can be calculated by the following equation:

$$\eta = -10 \times \log_{10}\left[\frac{1}{n} \times \sum_{i=1}^{n} y_i^2\right], \tag{21}$$

where η is amount of S/N ratio, n is the number of observations in the experiment, and y_i is value of characteristic for ith experiment. The algorithm's stages are as follows:

1. Choosing and assessing interfaces between factors;
2. Defining levels of factors;
3. Picking the suitable orthogonal group;
4. Allocating factors and their interfaces to every column of the orthogonal group;
5. Grasping the experiments;
6. Analyzing the results.

Khaw et al. [86] stated that Taguchi method considers orthogonal groups as a mathematical implement that study the smallest number of experiments with a large number of parameters. Through the Taguchi experimental design, quality increases in a different viewpoint. The method was useful in this study to decrease the number of experiments. For GA, there were three factors and all of them had five levels. Also, for ACO, there were five factors and all of them had five levels. Therefore, we use the L25 (L25, all five-levels) orthogonal group to decrease the number of experiments. This intended that only 25 experiments were necessary to grasp a decision. To find the truth, experiments were repeated five times (i.e., $25 \times 5 = 125$) for each problem.

5.3.1. An application: parameter tuning for GA and ACO

In this section, the offered stages above are followed to demonstrate the effectiveness of the method. The stages of application are as follows:

- Stage 1: We considered parameters P_c, P_m, and number of population for GA, and also considered parameters q, α, and β evaporation parameter, and number of population for ACO as the main parameters;
- Stage 2: The parameters and their values were named 'factors' and 'levels', respectively. The factors and their levels of GA and ACO are summarized in Table 1;
- Stage 3: In our problem, the factors had five levels. Hence, a L25 set was well suitable for our examination. Arrangement of orthogonal groups for parameter design is seen in the literature [87];
- Stage 4: The selected L25 design set for GA and ACO was displayed in Table 2, where the last column shows the total cost of 30 jobs problems;
- Stage 5: In this stage, we ran each test problem of Table 2 by MATLAB coding and recorded results of total cost for 30 jobs problems. Effective factors

Table 1. Parameters (factors) and their levels for GA and ACO.

Algorithm	Factors	Levels				
		1	**2**	**3**	**4**	**5**
GA	P_c	0.75	0.8	0.85	0.9	0.95
	P_m	0.006	0.007	0.008	0.009	0.01
	Number of population	80	90	100	110	120
ACO	q	0	0.25	0.5	0.75	1
	α	0.5	1	1.5	2	3
	β	1	2	3	5	10
	Evaporation parameter	0	0.25	0.5	0.75	1
	Number of population	80	90	100	110	120

Table 2. L25 design set for GA and ACO for 30 jobs.

Experiment number	GA factors			GA cost	ACO factors					ACO cost
	p_c	p_m	Number of population		q	α	β	Evaporation parameter	Number of population	
1	0.75	0.006	80	944.61	0	0.5	1	0	80	752.60
2	0.75	0.007	90	834.75	0	1	2	0.25	90	761.33
3	0.75	0.008	100	878.00	0	1.5	3	0.5	100	752.86
4	0.75	0.009	110	886.38	0	2	5	0.75	110	753.05
5	0.75	0.01	120	829.92	0	3	10	1	120	756.11
6	0.8	0.006	90	891.21	0.25	0.5	2	0.5	110	759.93
7	0.8	0.007	100	824.64	0.25	1	3	0.75	120	760.41
8	0.8	0.008	110	930.78	0.25	1.5	5	1	80	752.97
9	0.8	0.009	120	866.55	0.25	2	10	0	90	753.25
10	0.8	0.01	80	815.92	0.25	3	1	0.25	100	752.44
11	0.85	0.006	100	831.29	0.5	0.5	3	1	90	752.94
12	0.85	0.007	110	773.83	0.5	1	5	0	100	752.77
13	0.85	0.008	120	837.63	0.5	1.5	10	0.25	110	752.52
14	0.85	0.009	80	776.86	0.5	2	1	0.5	120	753.05
15	0.85	0.01	90	771.17	0.5	3	2	0.75	80	752.53
16	0.9	0.006	110	876.50	0.75	0.5	5	0.25	120	752.51
17	0.9	0.007	120	819.13	0.75	1	10	0.5	80	752.31
18	0.9	0.008	80	827.71	0.75	1.5	1	0.75	90	752.33
19	0.9	0.009	90	871.83	0.75	2	2	1	100	752.44
20	0.9	0.01	100	873.13	0.75	3	3	0	110	752.18
21	0.95	0.006	120	861.66	1	0.5	10	0.75	100	752.81
22	0.95	0.007	80	871.53	1	1	1	1	110	755.78
23	0.95	0.008	90	879.06	1	1.5	2	0	120	752.42
24	0.95	0.009	100	860.42	1	2	3	0.25	80	752.45
25	0.95	0.01	110	846.43	1	3	5	0.5	90	753.16

and levels of the effectiveness were established using Signal-to-Noise ratios, (S/N), which were selected as "the lowest value is the best" in Minitab software;

- Stage 6: In the last stage, analyses of the output of Taguchi method in Minitab software displayed that tuned parameters of GA and ACO are in accordance with those in Table 3. Also, S/N ratios criteria for 30 jobs problems by ACO are presented in Figure 5.

Table 3. Tuned parameters of GA and ACO by Taguchi method .

Total number of jobs	GA			ACO				
	P_c	P_m	Number of population	q	α	β	Evaporation parameter	Number of population
10	0.85	0.01	120	0.25	0.5	1	0	80
15	0.75	0.007	110	0.75	1	5	1	90
20	0.75	0.01	120	0.75	0.5	2	0.75	100
25	0.75	0.01	90	1	1	3	1	100
30	0.85	0.007	120	0.75	1.5	5	0	100

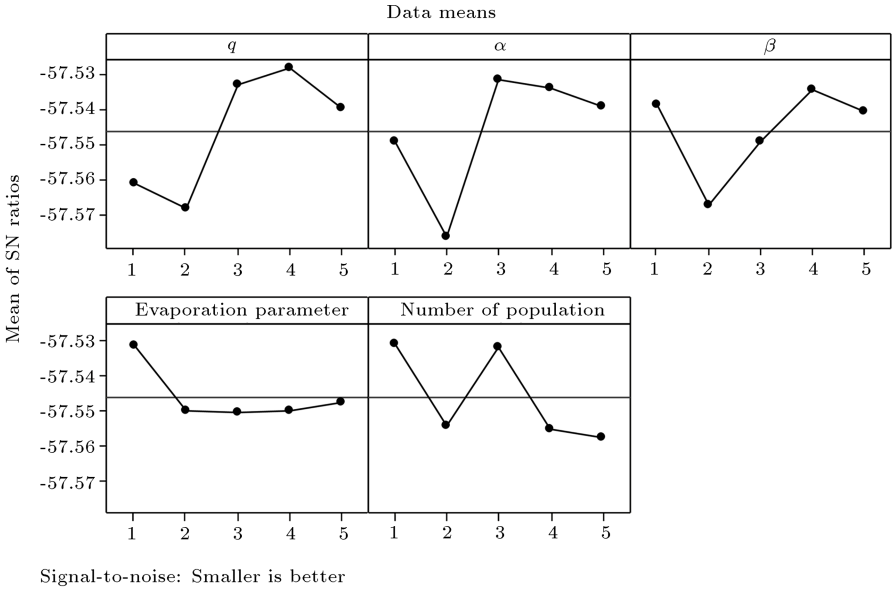

Figure 5. The main effects plot for SN ratio for ACO parameter (30 jobs problem).

5.4. The steps concerned with the solution method

- Step 1: In a specified test problem, determine the total cost of all jobs for the model by tuned GA and ACO;
- Step 2: Find the better algorithm for each test problem;
- Step 3: Make two hybrid algorithms (HACO & HGA) to determine the better and near-optimum solutions;
- Step 4: Determine the lower bound solution to the model using HACO;
- Step 5: Compare the results of the HACO with the lower bound in Step 4.

6. Numerical examples

In order to demonstrate the application of the proposed hybrid procedure and to study its performances, numerical examples are given in this section. The initial data of the examples for 30 jobs are given in Table 4. The initial parameter values for implementation of GA are as follows: probability of crossover (0.75-0.95), probability of mutation (0.005-0.01), population (80-120) and stopping criterion equal to 100 iterations. In addition, the initial parameter values for implementation of ACO are as follows: Number of ant (80-120), rate of evaporation (0-1), q (0-1), α (0-3), β (1-10), and stopping criterion equal to 100 iterations. It is noticeable from the literature that the parameters used in GA and ACO have robust effect on both result time and result quality. Hence, the GA and ACO parameters used were based on a Taguchi experimental design method mentioned in Subsection 5.3. In addition, five test problems with different number of jobs (10, 15, 20, 25, and 30 jobs) are used. All the test problems are solved on a notebook Intel core i5-4200 with 1.60 GHz CPU and 8 Gig RAM. Furthermore, the GA and ACO algorithms are coded using the MATLAB R2011a software.

The steps involved in the proposed procedure to solve the test problems are as follows:

- Step 1: In a given test problem, determine the total cost of all jobs for model by GA and ACO.

  In this step, each algorithm runs 20 times

Table 4. The initial data of the examples.

Jobs	Cell number	Batch size	Operation number	Unit processing time	Unit setup times	Total slots needed	Machine number
1		8	1	18, 12, 15	0.9, 0.6, 0.75	6, 3, 5	1, 2, 4
2		9	1	25, 21, 27	1.25, 1.05, 1.35	1, 2, 4	2, 1, 4
			2	24, 20	1.2, 1	5, 1	2, 1
3		13	1	26, 23	1.3, 1.15	2, 3	1, 2
			2	11, 15	0.55, 0.75	4, 6	2, 4
			3	8, 12, 9	0.4, 0.6, 0.45	4, 5,2	2, 1, 4
4		6	1	14, 13, 17	0.7, 0.65, 0.85	3, 6, 1	1, 2, 4
			2	19, 8, 15	0.95, 0.4, 0.75	4, 3, 1	1, 2, 4
5	1	9	1	22, 19, 16	1.1, 0.95, 0.8	6, 2, 6	2, 1, 4
			2	25, 17, 26	1.25, 0.85, 1.3	4, 5, 1	1, 2, 4
			3	9	0.45	3	4
6		10	1	16, 12	0.8, 0.6	2, 2	1, 4
			2	7, 13, 9	0.35, 0.65, 0.45	4, 5, 3	1, 2, 4
7		12	1	19, 19, 23	0.95, 0.95, 1.15	6, 4, 2	1, 2, 4
			2	13, 16, 10	0.65, 0.8, 0.5	6, 5, 1	1, 2, 4
			3	23, 26	1.15, 1.3	3, 3	2, 4
8		13	1	25, 18, 25	1.25, 0.9, 1.25	5, 4, 6	2, 1, 4
			2	7, 7, 15	0.35, 0.35, 0.75	1, 2, 6	1, 2, 4
			3	24, 31, 18	1.2, 1.55, 0.9	5, 4, 1	2, 4, 1
9		7	1	14, 13, 17	0.7, 0.65, 0.85	2, 1, 6	1, 5, 3
			2	19, 8, 15, 21	0.95, 0.4, 0.75, 1.05	4, 2, 5, 3	6, 1, 3, 5
10		8	1	25, 18, 25	1.25, 0.9, 1.25	2, 3, 4	6, 1, 5
			2	7, 15, 10	0.35, 0.75, 0.5	5, 6, 2	1, 3, 5
			3	24, 31, 18	1.2, 1.55, 0.9	3, 6, 4	6, 1, 3
11		9	1	25, 18, 25	1.25, 0.9, 1.25	1, 5, 5	5, 6, 1
			2	7, 7, 15	0.35, 0.35, 0.75	2, 6, 4	6, 3, 5
12		6	1	25, 18, 25	1.25, 0.9, 1.25	3, 3, 2	1, 6, 5
			2	7, 7, 15, 10	0.35, 0.35, 0.75, 0.5	6, 5, 1, 4	6, 3, 1, 5
	2		3	24, 31, 18	1.2, 1.55, 0.9	2, 5, 6	3, 1, 6
13		7	1	18, 12	0.9, 0.6	4, 3	5, 1
14		10	1	14, 13, 17	0.95, 0.95, 1.15	2, 1, 6	3, 6, 1
			2	10, 6	0.5, 0.3	3, 4	5, 6
15		11	1	12, 10, 13	0.6, 0.5, 0.65	3, 5, 4	5.3, 6
			2	5, 7	0.25, 0.35	6, 3	1, 5
			3	7, 9	0.35, 0.45	5, 1	6, 3
16		5	1	9,15	0.45, 0.75	2, 1	3, 2
			2	7, 8	0.35, 0.4	4, 3	2, 3
			3	10, 12	0.5, 0.6	6, 4	2, 3
17		4	1	8, 9	0.4, 0.45	3, 1	2, 3
			2	7, 8	0.35, 0.4	6, 5	3, 2
	3		3	13, 14	0.65, 0.7	6, 3	3, 2
18		8	1	14, 23	0.7, 1.15	2, 4	2, 3
			2	7, 12	0.35, 0.6	4, 5	3, 2
			3	10, 14	0.5, 0.7	6, 2	2, 3
19		12	1	14, 14	0.7, 0.7	3, 2	2, 3
			2	8, 11	0.4, 0.55	6, 4	3, 2
			3	5, 9	0.25, 0.45	1, 2	3, 2

Table 4. The initial data of the examples (continued).

Jobs	Cell number	Batch size	Operation number	Unit processing time	Unit setup times	Total slots needed	Machine number
			1	5, 8, 6, 10	0.25, 0.4, 0.3, 0.5	2, 5, 4, 2	8, 3, 7, 9
20		10	2	3, 4, 2, 4	0.15, 0.2, 0.1, 0.2	6, 1, 4, 3	7, 9, 3, 8
			3	2, 7	0.1, 0.35	3, 2	8, 3
			1	13, 14, 15, 2	0.65, 0.7, 0.75, 0.1	4, 5, 3, 6	7, 9, 3, 8
21		7	2	7, 5, 7, 6	0.35, 0.25, 0.35, 0.3	6, 1, 2, 1	9, 7, 8, 3
			3	2, 2, 5	0.1, 0.1, 0.3	6, 5, 2	3, 7, 9
			1	15, 13, 23, 11	0.75, 0.65, 1.15, 0.55	3, 4, 5, 6	3, 9, 7, 8
22	4	8	2	12, 13, 4, 5	0.6, 0.65, 0.2, 0.25	4, 4, 2, 1	7, 9, 3, 8
			3	4, 11	0.2, 0.55	1, 3	9, 7
			1	4, 11	0.2, 0.55	6, 5	7, 9
23		14	2	4, 13, 12	0.2, 0.65, 0.6	3, 4, 2	8, 3, 9
			3	11, 12, 23	0.55, 0.6, 1.15	1, 2, 3	7, 8, 9
			1	12, 13, 5	0.6, 0.65, 0.25	5, 6, 6	9, 3, 7
24		8	2	4, 11	0.2, 0.55	3, 4	8, 3
			3	5, 8	0.25, 0.4	5, 1	7, 9
			1	6, 7, 2	0.3, 0.35, 0.1	5, 3, 4	2, 7, 4
25		9	2	2, 5, 5	0.1, 0.25, 0.25	2, 6, 1	4, 2, 7
			3	2, 4, 4	0.1, 0.2, 0.2	2, 1, 4	7, 4, 2
			1	18, 21, 6	0.9, 1.05, 0.3	5, 3, 2	2, 7, 4
26		10	2	15, 6	0.75, 0.3	4, 1	2, 7
			3	6, 12	0.3, 0.6	4, 3	2, 4
			1	12, 14, 4	0.6, 0.7, 0.2	6, 3, 5	7, 4, 2
27		11	2	10, 4, 6	0.5, 0.2, 0.3	2, 1, 4	2, 7, 4
	5		3	4, 8	0.2, 0.4	2, 5	2, 4
			1	4	0.2	5	7
28		6	2	3, 16	0.15, 0.8	1, 4	2, 4
			3	4, 5	0.2, 0.25	3, 5	7, 4
			1	9, 9, 6	0.45, 0.45, 0.3	3, 2, 1	7, 2, 4
29		7	2	4, 6	0.2, 0.3	5, 4	7, 4
			3	13, 8	0.65, 0.4	6, 1	2, 7
			1	18	0.9	1	4
30		5	2	10, 5	0.5, 0.25	6, 5	7, 2
			3	3, 5	0.15, 0.25	3, 5	2, 4

for each test problem, where their minimum total costs, the least CPU times (s), and consumed power cost based on GA and ACO implementations are presented in Table 5;

- Step 2: The better algorithm is found by determining the percentage difference between their results. Based on the results given in Table 5, in all test problems, ACO is the better algorithm for total costs, the least CPU times (s), and consumed power cost. In terms of the total cost, ACO has always better performance, where its percentage improvements over GA are 0.00, 1.46, 2.3, 4.85, and 5.15 (average: 2.75) for 10, 15, 20, 25, and 30 jobs problems, respectively. Also, percentage improvements of CPU time (s) over GA are 19.36, 21.99, 15.28, 12.99, and 13.98 (average: 16.72) for 10, 15, 20, 25, and 30 jobs problems, separately. Furthermore, in terms of consumed power cost, ACO improvement percentages over GA are 0.00, 6.57, 7.15, 7.48, and 7.62 (average: 5.76) for 10, 15, 20, 25, and 30 jobs problems, respectively. The comparison results of the algorithms are presented in Figures 6-8. Also, all improvement trends are shown in Figure 9;

- Step 3: Make two hybrid algorithms (HACO & HGA) to determine the better, near-optimum solutions.

Regarding the results in Step 1, in this step, two hybrid algorithms are suggested to find a better

Table 5. Total cost results of all algorithms for all examples (Steps 1-5).

Total number of jobs	Total cost ($)		CPU time (s)		Power cost ($)		Hybrid total cost		Lower bounds	Diff.[a] with HACO	Percen.[b] penalty %	HACO detailed cost		
	ACO	GA	ACO	GA	ACO	GA	HGA	HACO				Perfor.[c] cost	Power cost	Total cost
10	349.23	349.23	14.74	18.28	23.07	23.07	349.23	349.23	348.47	0.76	0.22	326.16	23.07	349.23
15	512.11	519.70	17.27	22.14	38.10	40.78	517.07	511.82	510.14	1.68	0.33	473.74	38.08	511.82
20	572.36	585.85	22.35	26.38	45.73	49.25	582.29	571.83	569.00	2.83	0.50	526.36	4 5.47	571.83
25	691.02	726.26	27.18	31.24	57.37	62.01	698.43	670.02	665.52	4.5	0.68	614.74	55.28	670.02
30	775.88	818.04	32.42	37.69	65.75	71.17	796.49	752.03	741.61	10.42	1.40	688.24	63.79	752.03

[a]Diff.: Difference; [b]Percen.: Percentage; [c]Perfor.: Performance.

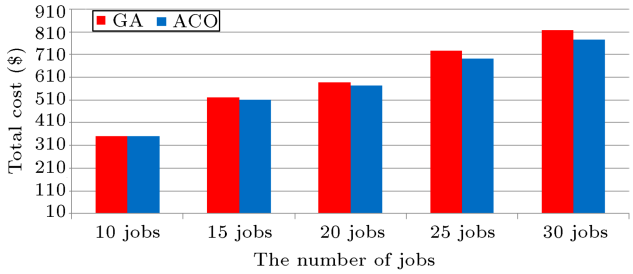

Figure 6. The total cost comparison of meta-heuristic algorithms (Steps 2).

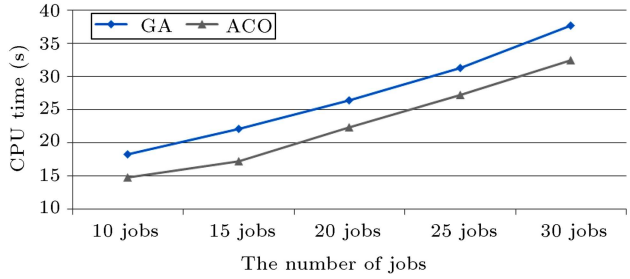

Figure 7. The CPU time comparison of meta-heuristic algorithms (Steps 2).

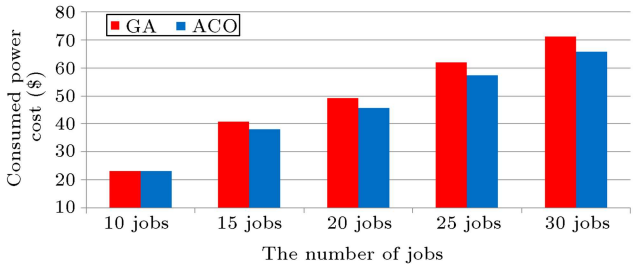

Figure 8. The consumed power cost of meta-heuristic algorithms (Steps 2).

near-optimum solution. We take the best outcomes of $(U_C, O_C, U_P, O_P, Y_j, X_{jogcml}, \text{AGV}_g)$ obtained by ACO for each test problem as an initial solution and input them into GA to make a Hybrid GA (HGA). Similarly, to make a hybrid ACO (HACO), GA is used to produce the best initial solutions. After that, running both hybrid algorithms 15 times for

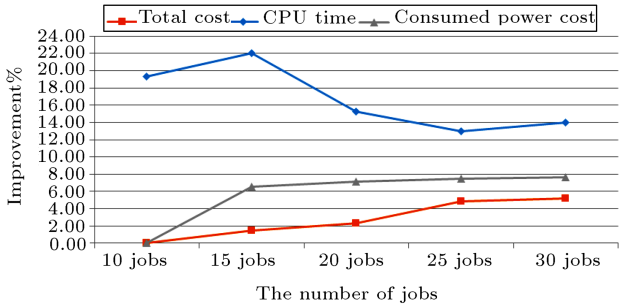

Figure 9. ACO improvement trends for total cost, CPU time, and power cost (Step 2).

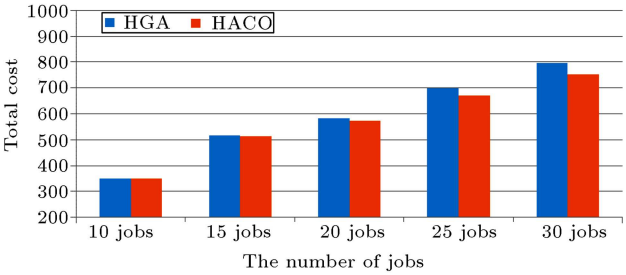

Figure 10. The total cost comparison of hybrid algorithms (Step 3).

each test problems, the minimum total costs and the detailed results of HACO are noted in Table 5. With respect to Table 5, HACO is absolutely the better hybrid algorithm. In addition, in terms of the total cost, HACO's improvement percentages over HGA's are 0.00, 1.01, 1.80, 4.06, and 5.58 (average: 2.49) or 10, 15, 20, 25, and 30 jobs problems, respectively. The comparison results by both hybrid algorithms and the improvement trends are shown in Figures 10 and 11, respectively;

- Step 4: Determine the lower bound solution to the model using HACO. To shed some light on the solution given by the presented hybrid algorithms, a required solution is compared with a lower bound. The lower bound is determined by solving the relaxed problem [88]. In this study, we ignored pert

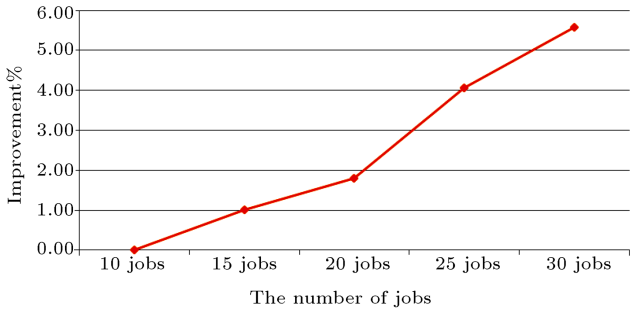

Figure 11. The HACO total cost improvement over HGA (Step 4).

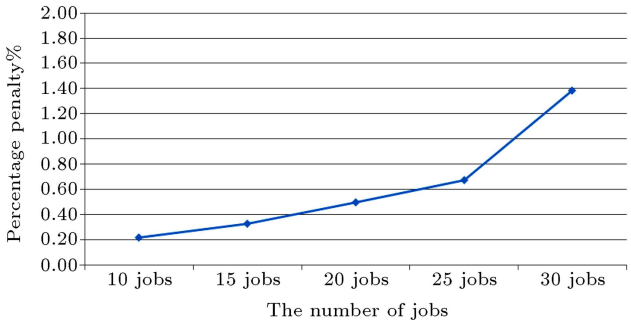

Figure 12. The percentage penalty of lower bound over HACO (Step 5).

percentage restriction and related costs. The lower bounds for the relaxed model of all test problems solved by an HACO are given in Table 5;

- Step 5: Compare the results of the HACO with the lower bound in Step 4. Now, one can settle the difference between the total costs of the hybrid solution with the lower bound. The difference can be concluded by:

$$\text{Difference} = \text{total cost of hybrid solution} - \text{lower bound}. \quad (22)$$

If the difference between the total costs obtained by HACO and the lower bound is small, then the difference among the proposed solution method and the unknown optimal solution should be small and the solution assumed by the proposed hybrid algorithm turns out to be the near-optimal solution since it is very close to the lower bound. If there is a large difference between the two solutions though, then this provides us with uncertainty about the effectiveness of the hybrid algorithm. Consequently, it is better to describe another comparison measure. The percentage penalty is a well-known measure of performance that is generally used. The percentage penalty is defined as follows:

$$\text{Percentage penalty} = \left(\frac{\text{difference}}{\text{lower bound}}\right) * 100\%. \quad (23)$$

Based on Eq. (23), if the percentage penalty measure is low, then the actual percentage difference between the solution obtained by HACO and the unknown optimal solution should be low [88]. Table 5 encompasses the comparison results of the proposed HACO with the lower bound. Furthermore, Figure 12 shows HACO percentage penalty of all test problems. Consistent with the solutions stated in Table 5, the minimum, maximum, and average percentage penalties are 0.22, 1.40, and 0.67, respectively. One can conclude that the solution given by the proposed hybrid algorithm turns out to be the near-optimal solution, because it is very close to the lower bound.

7. Conclusions and future research

7.1. Contributions

In this paper, an energy-conscious Cell Scheduling Problem (CSP) in a Flexibility Manufacturing System (FMS) was developed. In comparison to the model proposed by Abazari et al. [22] and Zhang et al. [42], the model works on Time-Of-Use (TOU) electricity tariffs and contains extra constraints such as total electric energy consumption, setup time available on each cell, part defect (pert) percentage, and the total number of AGV. We proposed a 5-step hybrid meta-heuristic method containing a Hybrid Ant Colony Optimization (HACO) and Hybrid Genetic Algorithm (HGA) to find a near-optimum solution to a Mixed Integer Linear Programming (MILP) model. The objective is to minimize the costs associated with the under-utilized cost, over-utilized cost, setup times cost, pert cost, process cost, and consumed power cost. Since there were no standards obtainable in the literature, a GA and an ACO were also developed for the solution. For parameter tuning of algorithms, we applied Taguchi experimental design method. In addition, to improve the quality of the results obtained by meta-heuristic algorithms, two hybrid algorithms (HGA and HACO) were proposed to solve the model. Furthermore, to confirm the proposed hybrid algorithm which works well, its outcomes were matched to lower bounds that were obtained by solving a relaxed model when pert percentage restriction and related costs were ignored using a HACO. Then, five numerical examples were presented to demonstrate the application of the proposed methodology. The results showed that the proposed hybrid procedure is able to find better and nearer-optimal solutions because they are very close to their lower bounds.

7.2. Limitations, theoretical, and managerial implications

In this study, for decreasing difficulty in the model, we ignored some issues and assumptions that we now recommend them to be considered for future studies in this area:

(a) Some of parameter of the model may be considered either as fuzzy or random. In this case, the model has either fuzzy or stochastic nature;

(b) Other meta-heuristic algorithms, such as Differential Evolution (DE), Imperialist Competitive Algorithm (ICA), and Particle Swarm Optimization (PSO), may also be applied to solve the model;

(c) Consideration of multiple objectives for optimization;

(d) Greenhouse gas (GHG) emissions can be added to energy-efficient model;

(e) Acceleration and deceleration of AGV movement can be considered;

(f) Different speeds of AGV movement with respect to filled or emptiness can be take into account;

(g) Automated storage and retrieval systems (AS/RS) for CSP warehouse can be considered;

(h) Sequence-Dependent Setup Time (SDST) can be studied.

References

1. Wemmerlov, U. and Hyer, N.L. "Cellular manufacturing in the U.S. industry: a survey of users", *International Journal of Production Research*, **27**(9), pp. 1511-1530 (1989).
2. Suer, G.A. "An algorithm to find the number of parallel stations for optimal cell scheduling", *Computers & Industrial Engineering*, **23**(1-4), pp. 81-84 (1992).
3. Selim, H.M., Askin, R.G., and Vakharia, A.J. "Cell formation in group technology: Review, evaluation, and directions for future research", *Computers & Industrial Engineering*, **34**(1), pp. 3-20 (1998).
4. Yu, J.J., Sun, S.D., Si, S.B., Yang, H.G., and Wu, X.L. "A study on the aviation manufacture cell scheduling based on adaptive ant colony algorithm", *Materials Science Forum*, **532-533**, pp. 1060-1063 (2007).
5. Saravanan, M. and Haq, A.N. "A scatter search method to minimize make-span of cell scheduling problem", *International Journal of Agile Systems and Management*, **3**(1-2), pp. 18-36 (2008).
6. Li, D.N., Wang, Y., and Xiao, G.X. "Dynamic parts scheduling in multiple job shop cells considering intercell moves and flexible routes", *Computers & Operations Research*, **40**(5), pp. 2007-2023 (2013).
7. Tang, J.F., Wang, X., Kaku, I., and Yung, K.L. "Optimization of parts scheduling in multiple cells considering intercell move using scatter search approach", *Journal of Intelligent Manufacturing*, **21**(4), pp. 525-537 (2010).
8. Schaller, J. "A comparison of heuristics for family and job scheduling in a flow-line manufacturing cell", *International Journal of Production Research*, **38**(2), pp. 287-308 (2000).
9. Hendizadeh, H., Faramarzi, H., Mansouri, S.A., Gupta, J.N.D., and Elmekkawy, T.Y. "Meta-heuristics for scheduling a flow shop manufacturing cell with sequence dependent family setup times", *International Journal of Production Economics*, **111**(2), pp. 593-605 (2008).
10. Lin, S.W., Ying, K.C., Lu, C.C., and Gupta, J.N.D. "Applying multi-start simulated annealing to schedule a flow line manufacturing cell with sequence dependent family setup times", *International Journal of Production Economics*, **130**(2), pp. 246-254 (2011).
11. Nagarjuna, N., Mahesh, O., and Rajagopal, K. "A heuristic based on multi-stage programming approach for machine loading problem in a flexible manufacturing system", *Robotics and Computer Integrated Manufacturing*, **22**, pp. 342-352 (2006).
12. Saravanan, M. and Noorul Haq, A.N. "Evaluation of scatter-search approach for scheduling optimization of flexible manufacturing systems", *International Journal of Advanced Manufacturing Technology*, **38**(5), pp. 978-986 (2008).
13. Blazevicz, J., Eiselt, H.A., Finke, G., Laporte, G., and Weglarz, J. "Scheduling tasks and vehicles in a flexible manufacturing system", *International Journal of Flexible Manufacturing System*, **4**(1), pp. 5-16 (1991).
14. Tanchoco, J.M.A. and Sinriech, D. "OSL-optimal single-loop guide paths for AGVS", *International Journal of Production Research*, **30**(3), pp. 665-681 (1992).
15. Sinriech, D. and Tanchoco, J.M.A. "The centroid projection method for locating pick-up and delivery stations in single-loop AGV systems", *Journal of Manufacturing Systems*, **11**(4), pp. 297-307 (1992).
16. Lengerke, O., Campos, A.M.V., Dutra, M.S., and Pinto, F.D.N.C. "Trajectories and simulation model of AGVs with trailers", *ABCM Symposium Series in Mechatronics*, **4**, pp. 509-518 (2010).
17. The Cadmus Group, "Regional lectricity emission factors final report" (1998).
18. Ulusoy, G., Sivrikaya-Serifoglus, F., and Bilge, U. "A genetic algorithm approach to the simultaneous scheduling of machines and automated guided vehicles", *Computers & Operations Research*, **24**(4), pp. 335-351 (1997).
19. Ali A. Pouyan, Heydar Toossian Shandiz, Soheil Arastehfar "Synthesis a Petri net based control model for a FMS cell", *Computers in Industry*, **62**, pp. 501-508 (2011).
20. Pach, C., Bekrar, A., Zbib, N., Sallez, Y. and Trentesaux, D. "An effective potential field approach to FMS holonic heterarchical control", *Control Engineering Practice*, **20**, pp. 1293-1309 (2012).
21. Leitão, P. and Restivo, F. "ADACOR: A holonic architecture for agile and adaptive manufacturing control", *Computers in Industry*, **57**(2), pp. 121-130 (2006).
22. Abazari, A.M., Solimanpur, M., and Sattari, H. "Optimum loading of machines in a flexible manufacturing

system using a mixed-integer linear mathematical programming model and genetic algorithm", *Computers & Industrial Engineering*, **62**, pp. 469-478 (2012).

23. Pach, C., Berger, T., Bonte, T., and Trentesaux, D. "ORCA-FMS: a dynamic architecture for the optimized and reactive control of flexible manufacturing scheduling", *Computers in Industry*, **65**(4), pp. 706-720 (2014).
24. Balaji, A.N. and Porselvi, S. "Artificial immune system algorithm and simulated annealing algorithm for scheduling batches of parts based on job availability model in a multi-cell flexible manufacturing system", *Procedia Engineering*, **97**, pp. 1524-1533 (2014).
25. Erdin, E.M. and Atmaca, A. "Implementation of an overall design of a flexible manufacturing system", *Procedia Technology*, **19**, pp. 185-192 (2015).
26. He, Y., Stecke, K., and Smith, M. "Robot and machine scheduling with state-dependent part input sequencing in flexible manufacturing systems", *International Journal of Production Research* (2016). DOI: 10.1080/00207543.2016.1161252
27. Solimanpur, M., Vrat, P., and Shankar, R. "A heuristic to minimize make-span of cell scheduling problem", *International Journal of Production Economics*, **88**(3), pp. 231-241 (2004).
28. Logendran, R., Mai, L., and Talkington, D. "Combined heuristics for bi-level group scheduling problems", *International Journal of Production Economics*, **38,** pp. 133-145 (1995).
29. Venkataramanaiah, S. "Scheduling in cellular manufacturing systems: a heuristic approach", *International Journal of Production Research,* **46**(2), pp. 429-449 (2008).
30. Tavakkoli-Moghaddam, R., Javadian, N., Khorrami, A., and Gholipour-Kanani, Y. "Design of a scatter search method for a novel multi-criteria group scheduling problem in a cellular manufacturing system", *Expert Systems with Applications,* **37**, pp. 2661-2669 (2009).
31. Lin, S.W., Ying, K.C., and Lee, Z.J. "Meta-heuristics for scheduling a non- permutation flow line manufacturing cell with sequence dependent family setup times", *Computers and Operations Research*, **36**, pp. 1110-1121 (2009).
32. Shirazi, B., Fazlollahtabar, H., and Mahdavi, I. "A six sigma based multi-objective optimization for machine grouping control in flexible cellular manufacturing systems with guide-path flexibility", *Advances in Engineering Software*, **41**, pp. 865-873 (2010).
33. Saadettin Erhan Kesen, Sanchoy K. Das, Zülal Güngör "A genetic algorithm based heuristic for scheduling of virtual manufacturing cells (VMCs)", *Computers & Operations Research*, **37**, pp. 1148-1156 (2010).
34. Lin, S.W., Ying, K.C., Lu, C.C., and Gupta, J.N.D. "Applying multi-start simulated annealing to schedule a flow line manufacturing cell with sequence dependent family setup times", *International Journal of Production Economics*, **130**(2), pp. 246-254 (2011).
35. Kia, R., Baboli, A., Javadian, N., Tavakkoli-Moghaddam, R., Kazemi, M., and Khorrami, J. "Solving a group layout design model of a dynamic cellular manufacturing system with alternative process routings, lot splitting and flexible reconfiguration by simulated annealing", *Computers & Operations Research*, **39**, pp. 2642-2658 (2012).
36. Batur, G.D., Karasan, O.E., and Akturk, M.S. "Multiple part-type scheduling in flexible robotic cells", *International Journal of Production Economics*, **135**(2), pp. 726-740 (2012).
37. Izui, K., Murakumo, Y., Suemitsu, I., Nishiwaki, S., Noda, A., and Nagatani, T. "Multiobjective layout optimization of robotic cellular manufacturing Systems", *Computers & Industrial Engineering*, **64**, pp. 537-544 (2013).
38. Boutsinas, B. "Machine-part cell formation using biclustering", *European Journal of Operational Research*, **230**, pp. 563-572 (2013).
39. Fazlollahtabar, H. and Jalali Naini, S.G. "Adapted Markovian model to control reliability assessment in multiple AGV manufacturing system", *Journal of Scientia Iranica*, **20**(6), pp. 2224-2237 (2013).
40. Yan, C.Z.J.T.C. "Job-shop cell-scheduling problem with inter-cell moves and automated guided vehicles", *Journal of Intelligent Manufacturing*, **26**, pp. 845-859 (2014).
41. Forghani, K. and Mohammadi, M. "A genetic algorithm for solving integrated cell formation and layout problem considering alternative routings and machine capacities", *Journal of Scientia Iranica*, **21**(6), pp. 2326-2346 (2014).
42. Zhang, H., Zhao, F., Fang, K., and Sutherland, J. "Energy-conscious flow shop scheduling under time-of-use electricity tariffs", *CIRP Annals - Manufacturing Technology*, **63**, pp. 37-40 (2014).
43. Li, Y., Li, X., and Gupta, J. "Solving the multi-objective flow line manufacturing cell scheduling problem by hybrid harmony search", *Expert Systems with Applications*, **42**, pp. 1409-1417 (2015).
44. Zohrevand, A.M., Rafiei, H., and Zohrevand, A.H. "Multi-objective dynamic cell formation problem: A stochastic programming approach", *Journal of Computers & Industrial Engineering*, **98**, pp. 323-332 (2016).
45. Majumder, A. and Laha, D. "A new cuckoo search algorithm for 2-machine robotic cell scheduling problem with sequence-dependent setup times", *Journal of Swarm and Evolutionary Computation*, **28**, pp. 131-143 (2016).
46. Gultekin, H., Akturk, M.S., and Karasan, O.E. "Bicriteria robotic operation allocation in a flexible manufacturing cell", *Computers & Operations Research*, **37**, pp. 779-789 (2010).
47. Tüysüz, F. and Kahraman, C. "Modeling a flexible manufacturing cell using stochastic Petri nets with fuzzy parameters", *Expert Systems with Applications*, **37**, pp. 3910-3920 (2010).

48. Naderi, B. and Azab, A. "Modeling and scheduling a flexible manufacturing cell with parallel processing capability", *CIRP Journal of Manufacturing Science and Technology*, **11**, pp. 18-27 (2015).

49. Yang, Y., Chen, Y., and Long, C. "Flexible robotic manufacturing cell scheduling problem with multiple robots", *International Journal of Production Research*, pp. 1-14 (2016)

50. Logendran, R., Carson, S., and Hanson, E. "Group scheduling in flexible flow shops", *International Journal of Production Economics*, **96**, pp. 143-155 (2004).

51. Logendran, R., de Szoeke, P., and Barnard, F. "Sequence-dependent group scheduling problems in flexible flow shops", *International Journal of Production Economics*, **102**, pp. 66-86 (2005).

52. Salmasi, N., Logendran, R., and Skandari, M. "Total flow time minimization in a flow shop sequence-dependent group scheduling problem", *Computers and Operations Research*, **37**, pp. 199-212 (2010).

53. Bruzzone, A.A.G., Anghinolfi, D., Paolucci, M., and Tonelli, F. "Energy-aware scheduling for improving manufacturing process sustainability: A mathematical model for flexible flow shops", *CIRP Annals - Manufacturing Technology*, **61**, pp. 459-462 (2012).

54. Mozdgir, A., Fatemi Ghomi, S.M.T., Jolai, F., and Navaei, J. "Three meta-heuristics to solve the no-wait two-stage assembly flow-shop scheduling problem", *Journal of Scientia Iranica*, **20**(6), pp. 2275-2283 (2013).

55. Jolai, F. and Abedinnia, H. "Consideration of transportation lags in a two-machine Flow shop scheduling problem", *Journal of Scientia Iranica*, **20**(6), pp. 2215-2223 (2013).

56. Jolai, F., Tavakkoli-Moghaddam, R., Rabiee, M., and Gheisariha, E. "An enhanced invasive weed optimization for make-span minimization in a flexible flow shop scheduling problem", *Journal of Scientia Iranica*, **21**(3), pp. 1007-1020 (2014).

57. Seidgar, H., Zandieh, M., and Mahdavi, I. "Bi-objective optimization for integrating production and preventive maintenance scheduling in two-stage assembly flow shop problem", *Journal of Industrial and Production Engineering*, **33**(6), pp. 404-425 (2016).

58. Fang, K., Uhan, N., Zhao, F., and Sutherland, J. "A new approach to scheduling in manufacturing for power consumption and carbon footprint reduction", *Journal of Manufacturing Systems*, **30**, pp. 234-240 (2011).

59. Dai, M., Tang, D., Giret, A., Salido, M., and Li, W.D. "Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm", *Robotics and Computer-Integrated Manufacturing*, **29**, pp. 418-429 (2013).

60. Zhang, R., Chang, P.C., and Wu, C. "A hybrid genetic algorithm for the job shop scheduling problem with practical considerations for manufacturing costs: Investigations motivated by vehicle production", *International Journal of Production Economics*, **145**(1), pp. 38-52 (2013).

61. Nourali, S. and Imanipour, N. "A particle swarm optimization-based algorithm for flexible assembly job shop scheduling problem with sequence dependent setup times", *Journal of Scientia Iranica*, **21**(3), pp. 1021-1033 (2014).

62. Yazdani, M., Zandieh, M., Tavakkoli-Moghaddam, R., and Jolai, F. "Two meta-heuristic algorithms for the dual-resource constrained flexible job-shop scheduling problem", *Journal of Scientia Iranica*, **22**(3), pp. 1242-1257 (2015).

63. Zhang, J., Yang, J., and Zhou, Y. "Robust scheduling for multi-objective flexible job shop Problems with flexible workdays", *Journal of Engineering Optimization*, **48**(11), pp. 1973-1989 (2016).

64. Hamta, N., Fatemi Ghomi, S.M.T., Tavakkoli-Moghaddam, R., and Jolai, F. "A hybrid meta-heuristic for balancing and scheduling assembly lines with sequence-independent setup times by considering deterioration tasks and learning effect", *Journal of Scientia Iranica*, **21**(3), pp. 963-979 (2014).

65. Khalili, S., Mohammadzade, H., and Fallahnezhad, M.S. "A new approach based on queuing theory for solving the assembly line balancing problem using fuzzy prioritization techniques", *Journal of Scientia Iranica*, **23**(1), pp. 387-398 (2016).

66. Kumar, A., Prakash, Tiwari, M.K., Shankar, R., and Baveja, A. "Solving machine-loading problem of a flexible manufacturing system with constraint-based genetic algorithm", *European Journal of Operational Research*, **175**, pp. 1043-1069 (2006).

67. Mukhopadhyay, S.K., Maiti, B., and Garg, S. "Heuristic solution to the scheduling problem in flexible manufacturing system", *International Journal of Production Research*, **29**, pp. 2003-2024 (1991).

68. Mukhopadhyay, S.K., Midha, S., and Krishna, V.M. "A heuristic procedure for loading problems in flexible manufacturing systems", *International Journal of Production Research*, **30,** pp. 2213-2228 (1992).

69. Mukhopadhyay, S.K., Singh, M.K., and Srivastava, R. "FMS loading: a simulated annealing approach", *International Journal of Production Research*, **36**, pp. 1629-1647 (1998).

70. Moreno, A.A. and Ding, F.Y. "Heuristic for the FMS loading and part type selection problems", *International Journal of Flexible Manufacturing Systems*, **5**(1), pp. 287-300 (1993).

71. Shanker, K. and Srinivasulu, A. "Some solution methodologies for loading problems in flexible manufacturing system", *International Journal of Production Research*, **27**(6), pp. 1019-1034 (1989).

72. Shankar, K. and Tzen, Y.J. "A loading and dispatching problem in a random flexible manufacturing system", *International Journal of Production Research*, **16**, pp. 383-393 (1985).

73. Diabat, A. "Hybrid algorithm for a vendor managed inventory system in a two-echelon supply chain", *European Journal of Operational Research*, **238**, pp. 114-121 (2014).

74. Sadeghi, J., Mousavi, S.M., Niaki, S.T.A., and Sadeghi, S. "Optimizing a multi-vendor multi-retailer vendor managed inventory problem: Two tuned meta-heuristic algorithms", *Knowledge-Based Systems*, **50**, pp. 159-170 (2013).

75. Chen, G., Govindan, K., and Yang, Z. "A method to reduce truck queuing at terminal gates: managing truck arrivals with vessel-dependent time windows", *International Journal of Production Economic*, **141**(1), pp. 179-188 (2013).

76. Sue-Ann, G., Ponnambalam, S.G., and Jawahar, N. "Evolutionary algorithms for optimal operating parameters of vendor managed inventory systems in a two-echelon supply chain", *Advances Engineering. Software*, **52**, pp. 47-54 (2012).

77. Pasandideh, S.H.R. and Niaki, S.T.A. "A genetic algorithm approach to optimize a multi-products EPQ model with discrete delivery orders and constrained space", *Applied Mathematics and Computation*, **195**, pp. 506-514 (2008).

78. Taleizadeh, A.A., Niaki, S.T.A., and Makui, A. "Multiproduct multiple-buyer single-vendor supply chain problem with stochastic demand, variable lead-time, and multi-chance constraint", *Expert Systems with Applications*, **39**, pp. 5338-5348 (2012).

79. Colorni, A., Dorigo, M., and Maniezzo, V. "Distributed optimization by ant colonies", *Proceedings of European Conference on Artificial Life*, Paris, France, pp. 134-142 (1991).

80. Colorni, A., Dorigo, M., and Maniezzo, V. "An investigation of some properties of an ant algorithm", *Proceedings of the Parallel Problem Solving from Nature Conference*, Brussels, Belgium, pp. 509-520 (1992).

81. Colorni, A., Dorigo, M., and Maniezzo, V. "Ant system for job-shop scheduling", *Belgian Journal of Operations Research, Statistics and Computer Science*, **34**, pp. 39-53 (1994).

82. Yang, W.H. and Tarng, Y.S. "Design optimization of cutting parameters for turning operations based on Taguchi method", *Journal of Materials Processing Technology*, **84**, pp. 122-129 (1998).

83. Davidson, M.J., Balasubramanian, K., and Tagore, G. "Experimental investigation on flow-forming of AA6061 alloy a Taguchi approach", *Journal of Materials Processing Technology*, **200**, pp. 283-287 (2008).

84. Du Plessis, B.J. and De Villiers, G.H. "The application of the Taguchi method in the evaluation of mechanical flotation in waste activated sludge thickening", *Resources, Conservation and Recycling*, **50**, pp. 202-210 (2007).

85. Taguchi, G., Chowdhury, S., and Wu, Y., *Taguchi's Quality Engineering Handbook*, Hoboken, NJ: Wiley. Talbi, El-Ghazali, Meta-heuristics, New York: Wiley (2005).

86. Khaw, J.F.C., Lim, B.S., and Lim, L.E.N. "Optimal design of neural networks using the Taguchi method", *Neurocomputing*, **7**, pp. 225-245 (1995).

87. Wu, Y. and Wu, A. *Taguchi Methods for Robust Design*, New York: The American Society of Mechanical Engineers (2000).

88. Cárdenas-Barrón, L.E., Trevino-Garza, G., and Wee, H.M. "A simple and better algorithm to solve the vendor managed inventory control system of multi-product multi-constraint economic order quantity model", *Expert Systems with Applications*, **39**, pp. 3888-3895 (2012).

Biographies

Mohammad Hemmati Far is a PhD student of Industrial Engineering at Science and Research Branch, Islamic Azad University, Iran. He has published in the areas of green supply chain, vendor managed inventory, fuzzy inventory control, and flexible cell scheduling problem.

Hassan Haleh, PhD, is an Assistant Professor of Industrial Engineering at Golpayegan University of Technology, Iran. He presently focuses his study on the areas of automated warehouses, supply chain, FMS, and CAD/CAM.

Abbas Saghaei, PhD, is an Associate Professor of Industrial Engineering at Science and Research Branch, Islamic Azad University, Iran. He currently focuses his research on the areas of statistical process control, design of experiments, and Six Sigma.