International Journal of Engineering

Journal Homepage: www.ije.ir

A Hybrid Genetic Algorithm for Integrated Production and Distribution Scheduling Problem with Outsourcing Allowed

L. Izadi, F. Ahmadizar*, J. Arkat

Department of Industrial Engineering, University of Kurdistan, Sanandaj, Iran

PAPER INFO

Paper history:
Received 07 June 2020
Received in revised form 13 July 2020
Accepted 03 September 2020

Keywords:
Dominance Properties
Hybrid Genetic Algorithm
Inventory
Outsourcing
Production Scheduling
Vehicle Routing

ABSTRACT

In this paper, we studied a new integrated production scheduling, vehicle routing, inventory and outsourcing problem. The production phase considers parallel machine scheduling including setup times with outsourcing allowed and the distribution phase considered batch delivery by a fleet of homogenous vehicles with respect to holding cost of completed jobs. The objective of the Mixed Integer Linear Programming (MILP) formulated model is to minimize the total costs including production, outsourcing, holding, tardiness and distribution fixed and variable costs. Due to the nondeterministic polynomial time (Np)-hardness of the problem, we derive a number of dominance properties for the optimal solution and combine them with a Genetic Algorithm (GA) to solve the problem. To assess the efficiency and effectiveness of the proposed hybrid algorithm, we conduct the computational study on randomly generated instances. Sensitivity analyses showed the impacts of the parameters on the objective function were incorporated. In order to evaluate the significance of the differences among the results obtained by GA and GADP one-tailed paired *t* tests were performed and interval plots were depicted.

***doi**: 10.5829/ije.2020.33.11b.19*

1. INTRODUCTION

Scheduling is a decision-making process in many manufacturing and service sectors, such as production, health, tourism, hospitality, and transportation [1]. Distribution is one of the elements of marketing which can be defined as the process of moving products from a manufacturer to the final consumer. In the other words, the most important duty of the distribution channel management is to send goods in the right place at the right time to the customers [2].

Recently, the global competition in the marketplace has forced the companies to minimize the amount of inventory required through the supply chain. However, the companies should be still responsive to the requirements of the customers. Minimized inventory can make an effective interaction between production and distribution processes leading to applicable effectiveness of integrated models [3, 4]. Outsourcing makes the manufacturers able to efficiently deal with fluctuations without keeping a high production or inventory capacity [5]. Deciding whether to produce in-house or employ an external supplying source has always been a fundamental challenge in manufacturing industries [6, 7].

In recent years, researchers have attended the integrated production scheduling and vehicle routing problem (IPSVRP). As it will be discussed in detail in the literature review, in most of researches in the literature handling the demand fluctuations as well as overcoming the capacity limitation by outsourcing option to meet customers' requests were ignored. Inventory issue and holding costs have become very important in the current competitive world. However, inventory of completed jobs and related holding cost before departure of delivery vehicles are rarely considered in integrated scheduling production and vehicle routing problem studies. In this paper, integrated production scheduling, inventory and vehicle routing with outsourcing problem in the parallel machine environment is studied. There are customers that order special products from a company that have parallel identical production lines. The products are processed in one stage on the one of the in-house lines or outsourced

*Corresponding Author Institutional Email: *f.ahmadizar@uok.ac.ir* (F. Ahmadizar)

Please cite this article as: L. Izadi, F. Ahmadizar, J. Arkat, A Hybrid Genetic Algorithm for Integrated Production and Distribution Scheduling Problem with Outsourcing Allowed, International Journal of Engineering (IJE), IJE TRANSACTIONS B: Applications Vol. 33, No. 11, (November 2020) 2285-2298

to a subcontractor. After completing, the goods are delivered to the costumers by vehicles. In order to minimize the fixed costs and increase the percentage of vehicle use, the goods are batched and then delivered to the customers using vehicles in some tours. The objective is to minimize the total costs, including production, holding, outsourcing, transportation and tardiness penalty costs.

To the best of our knowledge, it is the first time that this problem is being investigated in the literature to reduce the research gap. Two well-known solution approaches were utilized in order to solve the problem, i.e. exact and heuristic methods (due to Np-hardness of the problem [8]). For the former, a Mixed Integer Linear Programming (MILP) is developed and for the latter, a hybrid meta-heuristic incorporating the dominance properties with a Genetic Algorithm (GA) is proposed. The numerical experiments are examined to validate the MILP. The computational study is conducted for evaluating the efficiency and effectiveness of the proposed hybrid algorithm. The performance of the proposed algorithm to find the optimality is verified by MILP for small and medium size instances. Afterward, the capability of the proposed algorithm for solving the considered problem for the large instances is provided comparing to MILP in computational time. After vast numerical experiments, it can be concluded that proposed algorithm is capable to find optimal or near-optimal solution in comparison to MILP with less computational time and more quality.

The main contributions of this study are specified as follow:

- In order to be more responsive to the customer requests, outsourcing option is addressed in the current problem.
- In order to handle inventory issues, the holding cost incurred to the system is considered in this research.
- To the best of our knowledge, it is the first time that both outsourcing option and inventory are integrated in the related literature.
- To solve the problem optimally, a MILP is developed and a new hybrid GA is proposed. Computational study is conducted to evaluate the performance of proposed algorithm.
- A hybrid algorithm is used in context of integrated production scheduling and vehicle routing problem. The derived dominance properties play a strong neighbour-hood search role for GA.

2. LITERATURE REVIEW

In this section, the related literature on IPSVRP is reviewed.

Van Buer et al. [9] introduced integrated production scheduling on single machine and vehicle routing problem including the sufficient number of homogeneous vehicles with general order sizes.

In different researches, IPSVRP including batch delivery to multiple customers was investigated using adequate number of homogeneous vehicles with equal order sizes and single, parallel and bundling machine configuration [10, 11].

Several studies have been conducted on IPSVRP to multiple customers by batch delivery using limited number of homogeneous vehicles with single machine configuration [12–14].

Chen et al. [15] studied IPSVRP with time windows and stochastic demands at retailers for perishable food in order to maximize the expected total profit of the supplier. Besides, to deal with the problem, a combined algorithm of a constrained Nelder–Mead method and a heuristic was introduced.

Ullrich [16] investigated IPSVRP with multiple tours allowed for each vehicle to minimize the total tardiness. The integrated problem includes two sub-problems, parallel machines scheduling with ready times as well as the delivery of completed jobs with a fleet of heterogeneous vehicles. A genetic algorithm approach proposed to tackle the integrated problem indicates that the solutions of integrated problem are much more suitable than merged solutions of two sub-problems.

Amorim et al. [8] dealt with IPSVRP in parallel machine environment with set-up times comparing batching versus lot sizing for make-to-order production strategy in the cases with perishability of products. Results showed that lot sizing can lead to better solutions in comparison with batching.

Belho-Filho et al. [17] proposed an adaptive large neighborhood search (ALNS) framework to solve the problem of integrated lot sizing and scheduling and vehicle routing. Other characteristics of the problem are perishability nature of goods, parallel machine environment and sequence dependent setup times- are incurred because of reconfiguration of equipment for production different products. The objective is to minimize the sum of production costs, consist of setup and production cost, as well as distribution costs, including fixed cost of vehicle usage and distance proportional costs.

Kang et al. [18] studied IPSVRP with multiple vehicles and flow-shop machine configuration and outsourcing allowed in various stages in the manufacturing process. The objective is to minimize the total costs including production cost depending on various product types, factory setup cost and different outsourcing factories as well as transportation cost with multiple vehicles. To deal with the problem in large sizes, an efficient GA is proposed.

In order to minimize the total customer waiting time and the vehicle delivery cost, Li et al. tackled a multi-objective IPSVRP with a single machine and multiple vehicles [19]. Applying the elite strategy, they solved the problem using non-dominated genetic algorithm.

To minimize required time to process and deliver all customer demands, Karaoglan and Kesen [20] investigated an integrated single machine scheduling and capacitated vehicle routing problem for single time-sensitive product. They developed a branch-and-cut (B&C) algorithm applying several valid inequalities in order to improve lower bounds besides using a simulated annealing approach based local search to enhance upper bounds to solve the problem [20].

Devapriya et al. [21] investigated the integrated production scheduling and vehicle routing problem for perishable products. The limited capacity vehicles can be utilized multiple times in planning horizon. The problem tackled by heuristics based on evolutionary algorithms is modelled to determine production schedule, fleet size, and vehicle's route at minimum cost [21].

Lacomme et al. [22] evaluated the IPSVRP including specific capacity constraints, limited lifespan of goods and special case of a single vehicle. A greedy randomized adaptive search procedure (GRASP) with an evolutionary local search (ELS) is developed to tackle the considered problem. Besides, a new set of instances with multiple vehicles is introduced for further research [22].

Tamannaei and Rasti-Barzoki [23] investigated the IPSVRP to minimize the total weighted tardiness, fixed and variable transportation costs. A Branch-and-Bound (B&B) based exact procedure and a meta-heuristic genetic algorithm (GA) were proposed to solve the problem [23].

Tavares-Neto and Nagano [24] dealt with integrated scheduling production, inventory, and distribution problem aiming to integrate a scheduling of parallel machines considering sequence dependent setup time using a single vehicle with multiple routes for delivery. In order to solve the problem two new algorithms, i.e. an Iterated Greedy technique based improvement heuristic and a constructive heuristic, are proposed. Comparing the obtained results of the algorithms with a Mixed-Integer Programming model and an adapted Genetic Algorithm showed that the former can lead to better results [24].

Mohammadi et al. [25] studied an integrated flexible job-shop scheduling-vehicle routing problem with a time window. They modelled the problem as a novel bi-objective mixed integer, in which the first objective function tries to minimize a sum of the production and distribution scheduling costs, and the second objective function aims to minimize a weighted sum of delivery earliness and tardiness. They considered a furniture manufacturing company producing customized goods as a case study. In this study, an Ɛ-constraint method is applied to solve small-sized real data and medium- and large-sized problems are solved by a Hybrid Particle Swarm Optimization algorithm [25].

Mousavi et al. [26] addressed production and air transportation scheduling problem with time windows for the due date aimed to minimize the total supply chain costs. They developed four algorithms (i.e., SA, GA, PSO and DPSO) and hybrid variable neighborhood search–simulated annealing to solve different size generated problems [26].

Literature studies have indicated that a few articles addressed inventory cost associated with integrated production and distribution system with batch delivery. In addition, the outsourcing option in manufacturing stage is often neglected, however the effective impact of outsourcing on reducing investment requirements and improving the response to customers' demands is shown in the relative literature. To the best of our knowledge, among the literature of integrated production scheduling and vehicle routing problem reviewed, only one paper has considered inventory and holding cost specially and only one paper has addressed outsourcing option while none of them regards both inventory and outsourcing simultaneously. In this paper, we consider holding cost and outsourcing option in IPSVRP for the first time. Meanwhile, the proposed hybrid meta-heuristic algorithm, i.e. dominance properties incorporated with genetic algorithm, is being newly used in the related literature.

3. PROBLEM DEFINITION

In this paper, we address coordination between production scheduling with outsourcing allowed and vehicle routing problem. There is a factory that receives orders containing P types of products from customers. Each type of products is processed in one production stage. Orders are processed on either in-house m identical parallel lines or single subcontractor's line. Sequence-dependent setup times satisfying the triangle inequality - between producing different products- are taken into account on in-house lines. Setup cost is calculated by multiplying setup times by the coefficient of setup cost. The jobs are outsourced with outsourcing cost oc_{ij} and lead time l_{ij}; after processing, outsourced jobs are returned to the factory.

There are f customers in different geographic locations ordering a set of products. Customers' demand has to be delivered in strict due dates in some batches by routing with a fleet of homogeneous vehicles. Each vehicle starts the route from the depot of the factory and after visiting associated customers it returns to the depot of the factory. If customers' orders are delivered after associated due date, tardiness penalty will be incurred. Tardiness penalty is calculated by multiplying tardiness by penalty coefficient. Transportation cost is composed of two parts: fixed cost -usage cost of vehicle- and variable cost-which is proportional to transportation time.

Because of batch delivery form, a completed job of a batch must be kept in the depot until other jobs in the same batch are completed. In the other words, the delivery is carried out when all the jobs in the batch are completed. The incurred holding cost can be calculated by multiplying holding time by holding cost rate for each job. The aim is to find a joint production and distribution schedule with minimum total cost of setup, holding, outsourcing, transportation and tardiness.

Assumptions of this problem are written as follows:

- Batch is a shipment containing one or more customers' order delivered by one vehicle.
- Each customer can order one or more products.
- Vehicles are available in time zero in factory.
- Capacity of each vehicle is greater than the total demand of each customer.
- Each customer is visited only once.

3. 1. Mathematical Formulation

In this section, the problem is formulated in Mixed Integer Linear Programming (MILP). The parameters, variables, and constraints of model are provided as follow:

Parameters:

j: Product indices.

i: Customer indices (*i=0, n+1* denoted production location).

k: Production line indices.

v: Vehicle indices.

ij: Product j ordered by customer i.

f: Number of customers.

n_i: Number of jobs ordered by customer i.

m: Number of production lines.

N: Number of all jobs.

P: Number of products.

V: Number of vehicles.

p_{ij}: Processing time of job ij.

α_i: Tardiness penalty coefficient for customer i.

β: Cost of per unit of setup time.

$set_{jj'}$: Setup time between product j and j'.

set_{0j}: Set up time for the first job j on each line.

$t_{ii'}$: Travel time from the location of customer i to the location of customer i'. (i=0, n+1 denoted facility location).

h_{ij}: Rate of holding cost for job ij (unit of cost in per unit of time).

F_v: Fixed cost associated with each vehicle v.

c^v : Cost per travelling unit time travelled by vehicle v.

dem_{ij}: Demand of job ij.

d_i: Due date of all jobs ordered by customer i.

cap_v: Capacity of vehicle v.

q: An enough large number.

oc_{ij}: Outsourcing cost of job ij.

l_{ij}: Lead time of outsourced job ij.

Variables:

C_{ij}: Completion time of job ij.

y_{kij} : A binary variable indicating whether job ij is processed first on machine k.

$x_{iji'j'}$: A binary variable indicating whether job ij is processed immediately before job $i'j'$ on the same machine.

u_{ij}: A binary variable indicating whether job ij is processed last on a machine.

H_{ij}: The time that job ij waits for completing other jobs belonging to its batch.

w_i^v: Binary variable which takes the value 1 if customer order i is delivered by vehicle v (*i=0, i=f+1* denote processing site).

$z_{ii'}^v$: A binary variable which takes the value 1 if customer i is visited immediately before customer i' by vehicle $v \in V$ (*i=0, i=f+1* denote processing site).

z_{0i}^v: Binary variable which takes the value 1 if customer i is visited first by vehicle v.

$z_{0,f+1}^v$: Binary variable which takes the value 1 if vehicle v remains in processing site without using.

S_v: Start time of tour of vehicle v.

T_i: Tardiness of jobs ordered by customer i.

D_i: Delivery date of all jobs ordered by customer i.

o_{ij}: A binary variable indicating whether job ij is outsourced.

$$\text{Min } \sum_{i=1}^{f} \alpha_i T_i + \sum_{i=1}^{f}\sum_{j=1}^{n_i}\sum_{\substack{i'=1 \\ i \neq i'}}^{f}\sum_{\substack{j'=1 \\ j \neq j'}}^{n_{i'}} \beta\, set_{jj'} x_{iji'j'} + \sum_{i=1}^{f}\sum_{j=1}^{n_i}\sum_{k=1}^{m} \beta\, set_{0j} y_{kij} + \sum_{i=1}^{f}\sum_{j=1}^{n_i} H_{ij} h_{ij} + \sum_{v=1}^{V} F_v (1 - z_{0,f+1}^v) + \sum_{v=1}^{V}\sum_{\substack{i=0 \\ i \neq i'}}^{f}\sum_{i'=1}^{f+1} c^v\, t_{ii'} z_{ii'}^v + \sum_{i=1}^{f}\sum_{j=1}^{n_i} oc_{ij} o_{ij} \tag{1}$$

Objective (1) is to minimize the sum of production, distribution, outsourcing and holding costs. The total production cost is composed of sequence-dependent set-up costs. The distribution cost is composed of tardiness penalties and fixed vehicle usage costs as well as variable travel time proportional costs. The outsourcing cost is incurred by outsourcing the jobs. The holding cost is associated with holding finished goods in the factory depot.

Scheduling constraints:

$$\sum_{i=1}^{f}\sum_{j=1}^{n_i} y_{kij} \leq 1, k = 1, \dots, m \tag{2}$$

$$\sum_{k=1}^{m} y_{ki'j'} + \sum_{i=1}^{f}\sum_{j=1}^{n_i} x_{iji'j'} + o_{i'j'} = 1, i' = 1, \dots, f, j' = 1, \dots, n_{i'},\ \ ij \neq i'j' \tag{3}$$

$$\sum_{i'=1}^{f}\sum_{\substack{j'=1 \\ ij \neq i'j'}}^{p} x_{iji'j'} + u_{ij} + o_{ij} = 1, i = 1, \dots, f, j = 1, \dots, n_i \tag{4}$$

$$C_{ij} \geq p_{ij} y_{kij}, i = 1, \dots, f; j = 1, \dots, n_i; k = 1, \dots, m, ij \neq i'j' \tag{5}$$

$$C_{ij} \geq C_{i'j'} + p_{ij} + set_{j'j} - q(1 - x_{i'j'ij}), i, i' = 1, \ldots, f; j = 1, \ldots, n_i; j' = 1, \ldots, n_{i'}; ij \neq i'j' \quad (6)$$

$$C_{ij} \geq l_{ij} o_{ij}, i = 1, \ldots, f, j = 1, \ldots, n_i \quad (7)$$

Constraint (2) ensure there is at most one job which is the first to be processed on line *k*. Constraint (3) state each job is either processed after another one on a line or the first to be processed or outsourced. Constraint (4) state each job is either the last to be processed or precedes another job on a line or outsourced and Constraints (5), (6), (7) define the completion time of job *ij*.

Vehicle routing constraints:

$$\sum_{v=1}^{V} w_i^v = 1, i = 1, \ldots, f \quad (8)$$

$$w_0^v \geq w_i^v, v = 1, \ldots, V; i = 1, \ldots, f \quad (9)$$

$$\sum_{\substack{i=0 \\ i' \neq i}}^{f} z_{ii'}^v = w_{i'}^v, v = 1, \ldots, V, i' = 1, \ldots, f \quad (10)$$

$$\sum_{\substack{i=1 \\ i' \neq i}}^{f+1} z_{i'i}^v = w_{i'}^v, v = 1, \ldots, V; i' = 1, \ldots, f \quad (11)$$

$$c_v \geq \sum_{i=1}^{f} \sum_{j=1}^{n_i} dem_{ij} w_i^v, v = 1, \ldots, V \quad (12)$$

Constraint (8) assign each customer to exactly one vehicle and one tour. Constraint (9) guarantee a utilized vehicle that must start its trip from production facility. Constraints (10) and (11) explain the vehicle, that visits customer i', travels either from another customer or from the processing site to the location of customer. Following the service, the vehicle can come-back to the processing site or deliver the other customer's order. Constraint (12) ensure that the number of jobs loaded in each vehicle does not exceed the capacity of vehicle.

Integration constraints:

$$S_v \geq C_{ij} - q(1 - w_i^v), i = 1, \ldots, f, j = 1, \ldots, n_i, v = 1, \ldots, V \quad (13)$$

$$D_i \geq D_{i'} + t_{i'i} - q(1 - z_{i'i}^v), i, i' = 1, \ldots, f, i \neq i', v = 1, \ldots, V \quad (14)$$

$$D_i \geq S_v + t_{0i} - q(1 - w_i^v), i = 1, \ldots, f, v = 1, \ldots, V \quad (15)$$

$$T_i \geq D_i - d_i, i = 1, \ldots, f \quad (16)$$

$$H_{ij} \geq S_v - C_{ij} - q(1 - w_i^v), i = 1, \ldots, f, j = 1, \ldots, n_i, v = 1, \ldots, V \quad (17)$$

$$T_{ij}, S_v, C_{ij}, H_{ij}, D_i \geq 0, i = 1, \ldots, f, j = 1, \ldots, n_i \; v = 1, \ldots, V \quad (18)$$

$$x_{ij,i'j'}, y_{kij}, z_{ii'}^v, o_{ij} \in \{0,1\} \quad (19)$$

Constraint (13) guarantee that the starting time of vehicle *v* is greater than the completion time of the jobs delivered by this vehicle. Constraint (14) say that delivery date of customer order *i* is greater than that of previous customer (production site) plus traveling time between two customer locations. Constraint (15) state that delivery date of jobs ordered by customer *i* is greater than the starting time of associated vehicle plus time distance between production site and location of customer *i*. Constraint (16) define the tardiness of orders of customer *i*. Constraint (17) explain the holding time of job *ij* in the factory and Constraints (18) and (19) define the variables.

4. DOMINANCE PROPERTIES

In this section, three lemmas, two corollaries, and five properties for the problem are introduced. Lemma 3 and property 2 that are derived from literature [27] and corollary 2 is derived from literature [28] are applied in production scheduling section in this research.

Lemma 1: The in-house jobs constructing a batch are not essentially processed on each line in a continuous way in the optimal solution.

Property 1: Given a solution in which a line after processing job ij belonging to batch *v* is idle for period time IT. If $S_v \geq C_{ij} + IT'$, where $0 < IT' \leq IT$, the processing of job ij must be postponed to complete at $C_{ij} + IT'$.

Corollary 1: In the optimal solution, each production line is not idle after processing the job, except the last job of a batch.

Lemma 2: The optimal solution is not essentially in non-delay form.

Proof: The objective function of the problem is composed of five parts including set-up cost, holding cost, outsourcing cost, tardiness cost and transportation cost. Tardiness cost is a regular performance measure that is non-decreasing in completion time. Set-up cost, outsourcing cost and transportation cost are not dependent on completion time however, holding cost is decreasing in completion time; if the completion time of a job (except the last job) belonging to a batch increases, then it becomes closer to departure time of the vehicle and consequently resulting in the shortened holding time. Therefore, the objective function of this article is not regular. Supposing a non-delay solution, any idle time inserted before processing a job that will lead to the increased completion time of the job as well as increased probable next jobs on the same line besides, the holding cost is decreased and consequently leading to decreased objective function. As a result, it can be concluded that this unnecessary idle time may improve the solution.

Property 2 Given a solution in which job ij belonging to batch *v* precedes immediately with job $i'j$ on the same line. If job ij is not the last job of the batch, $h_{i'j} p_{ij} < h_{ij} p_{i'j}$ and $S_v - C_{ij} \geq p_{i'j}$, the two jobs should be interchanged.

Proof: Suppose schedule I is the solution in which job $i'j$belonging to batch v precedes immediately with job ij on the same line and the conditions mentioned above is satisfied. From corollary 1 there is no idle time between processing of these two jobs. Now, suppose schedule II in which jobs ij and $i'j$ are interchanged. Under solution II, the completion times of ij and $i'j$ are different from those in schedule I but, all the other jobs finish at the same time as in schedule I. The only difference between the two solutions lies in the holding cost of jobs ij and $i'j$.

Considering ST the time point at which job ijstarts in schedule I and job $i'j$starts in schedule II, the difference in cost of schedule I and II is as follow:

$I: h_{i'j}(S_v - ST - p_{i'j}) + h_{ij}(S_v - ST - p_{i'j} - p_{ij})$

$II: h_{ij}(S_v - ST - p_{ij}) + h_{i'j}(S_v - ST - p_{ij} - p_{i'j})$

After taking the differences of I and II, the following expression is obtained:

$I - II: h_{i'j}p_{ij} - h_{ij}p_{i'j} > 0$

Thus, the interchanging is cost effective and schedule II dominates schedule I and the proof is completed.

Property 3: Given a solution in which job $i'j'$ from batch v, that is not the last job of the batch, immediately precedes with job ij. If $S_v - C_{i'j'} \geq setup_{j'j} + p_{ij}$ and $h_{i'j'}(setup_{j'j} + p_{ij}) \geq h_{ij}(p_{i'j'})$, the two jobs should be interchanged.

Proof: Suppose a solution in which job $i'j'$ from batch v is not the last job of the batch and immediately precedes with job ij (schedule I). From corollary 1 there is no idle time between processing of these two jobs. Assume that $S_v - C_{i'j'} \geq setup_{j'j} + p_{ij}$ (without changingS_v after interchanging the two jobs), and $h_{i'j'}(setup_{j'j} + p_{ij}) \geq h_{ij}(p_{i'j'})$ as well as $setup_{j''j'} + setup_{j'j} \geq setup_{j''j}$from triangular setup times assumption. Now, suppose schedule II in which jobs ij and $i'j$ are interchanged. Under solution II, the completion times of ij and $i'j$ are different from those as in schedule I but all the other jobs finish at the same time as in schedule I. Obviously, the differences between the two solutions are in the holding cost for jobs ij and $i'j$ as well as setup cost; these differences are as follow:

$I: \beta.(setup_{j''j'} + setup_{j'j}) + h_{i'j'}(setup_{j'j} + p_{ij})$

$II: \beta.setup_{j''j} + h_{ij}(p_{i'j'})$

After taking the differences of I and II, based on assumptions, it is obtained that:

$I - II: \beta.(setup_{j''j'} + setup_{j'j} - setup_{j''j}) + h_{i'j'}(setup_{j'j} + p_{ij}) - h_{ij}(p_{i'j'}) \geq 0$

Thus, the interchanging is cost effective; schedule II dominates schedule I and the proof is completed.

Property 4: Given a solution in which two jobs ij and $i'j$ belong to batch v, job $i'j$ is outsourced and job ij is processed in-house. If $p_{ij} \geq p_{i'j}$, $l_{ij} \leq S_v$, $h_{i'j}(C_{ij} - l_{i'j} + p_{i'j} - p_{ij}) + h_{ij}(l_{ij} - C_{ij}) > 0$ and $oc_{ij} < oc_{i'j}$, job $i'j$ should be outsourced and job ij should be processed in-house.

Proof: Suppose schedule I is a solution in which two jobs ij and $i'j$ belong to batch v, job $i'j$ is outsourced and job ij is processed in-house. Assume that $p_{ij} \geq p_{i'j}$, $l_{ij} \leq S_v$, $h_{i'j}(C_{ij} - l_{i'j} + p_{i'j} - p_{ij}) + h_{ij}(l_{ij} - C_{ij}) > 0$ and $oc_{ij} < oc_{i'j}$. Now, construct schedule II in which job ij is outsourced and job $i'j$ is processed in-house replacing job ij. Schedule II must be constructed in the way that no change is occurred in S_v and completion times of the jobs after job $i'j$. Since $p_{ij} \geq p_{i'j}$, we insert an idle for time period $p_{ij} - p_{i'j}$ after job $i'j$. Afterwards, job $i'j$ and the jobs before $i'j$ belonging to the same line till the job which is the last job of its relative batch (called set A) are shifted to right for time period $p_{ij} - p_{i'j}$; this reduces the holding time and cost of the jobs. The differences in the cost of the schedule I and II are in the holding cost of the jobs and outsourcing cost as follow:

$I: OC_{i'j} + h_{ij}(S_v - C_{ij}) + h_{i'j}(S_v - l_{i'j}) + \sum_{kl \in A} h_{kl}(p_{ij} - p_{i'j})$

$II: OC_{ij} + h_{ij}(S_v - l_{ij}) + h_{i'j}(S_v - C_{i'j})$

Since $C_{i'j} = C_{ij} - p_{ij} + p_{i'j}$ and based on assumptions, the difference of schedule I and II is obtained as follow:

$I - II: (OC_{i'j} - OC_{ij}) + h_{i'j}(C_{ij} - l_{i'j} + p_{i'j} - p_{ij}) + h_{ij}(l_{ij} - C_{ij}) \geq 0$

Thus, schedule II dominates schedule I and the proof is completed.

Property 5: Given a solution in which job $i'j$ is outsourced and jobs ij and $i''j'$ are processed in the house in which job ij is produced immediately before (after) job $i''j'$. If $p_{i''j'} > p_{i'j}$, $l_{i''j'} < S_v$, $oc_{i'j} > oc_{i''j'}$ and $h_{i''j'}(l_{i''j'} - C_{i''j'}) + h_{i'j}(C_{i''j'} - p_{i''j'} + p_{i'j} - l_{i'j}) > 0$, job $i''j'$ should be outsourced and job $i'j$ should be processed in-house replacing job $i''j'$.

Proof: Suppose schedule I is a solution in which job $i'j$ is outsourced and two jobs ij and $i''j'$ are processed in-house in which job ij is processed immediately before job $i''j'$. Assume that $p_{i''j'} > p_{i'j}$, $l_{i''j'} < S_v$, $oc_{i'j} > oc_{i''j'}$ and $h_{i''j'}(l_{i''j'} - C_{i''j'}) + h_{i'j}(C_{i''j'} - p_{i''j'} + p_{i'j} - l_{i'j}) > 0$ as well as $setup_{jj'} + setup_{j'j''} > setup_{jj''}$ from triangular setup times assumption. Now, construct schedule II in which job $i''j'$ is outsourced and job $i'j$ is processed in-house replacing job $i''j'$. Without changing the completion time of jobs except $i'j$ in schedule II, insert an idle after $i'j$ for time period $p_{i''j'} - p_{i'j}$. Finally, to reduce holding time and consequently holding cost of the jobs, postpone job $i'j$ and the jobs before $i'j$ belonging to the same line until the job which is the last job of its relative batch (called set A). The differences in the cost of the schedule I and II are in the holding cost of the jobs, outsourcing cost and setup cost. The differences are as follow:

$I: \beta(setup_{jj'} + setup_{j'j''}) + h_{i''j'}(S_v - C_{i''j'}) + h_{i'j}(S_v - l_{i'j}) + oc_{i'j} + \sum_{kl\in A} h_{kl}(p_{i''j'} - p_{i'j})$

$II: \beta(setup_{jj''}) + h_{i''j'}(S_v - l_{i''j'}) + h_{i'j}(S_v - C_{i'j}) + oc_{i''j'}$

Since $C_{i'j} = C_{i''j'} - p_{i''j'} + p_{i'j}$, the difference of schedules I and II are obtained as follow:

$I - II: \beta(setup_{jj'} + setup_{j'j''} - setup_{jj''}) + h_{i''j'}(l_{i''j'} - C_{i''j'}) + h_{i'j}(C_{i''j'} - p_{i''j'} + p_{i'j} - l_{i'j}) + (oc_{i'j} - oc_{i''j'}) + \sum_{kl\in A} h_{kl}(p_{i''j'} - p_{i'j}) \ge 0$

Thus, the changes are cost effective; schedule II dominates schedule I and the proof is completed. If job ij is processed immediately after job $i''j'$ and job $i'''j''$ is processed before that, the proof is accomplished in the similar way.

5. SOLUTION APPROACH

To solve the problem efficiently, in this section a hybrid algorithm is proposed by incorporating a Genetic Algorithm (GA) with dominance properties, called GADP. The hybrid meta-heuristics with dominance properties are applied for solving some of previous scheduling or outsourcing problems; e.g. literatures [29, 30] proposed incorporating the dominance properties with genetic algorithm, literatures [5, 31] incorporated the dominance properties with ant colony algorithm.

Genetic algorithm [32–36] is one of the most well-known population based evolutionary meta-heuristics mimicking Darwinian principle of survival of the fittest, used for combinatorial problems. The initial population is generated by encoding the solutions of the considered problem to chromosomes. In the iterative steps, after selecting the parents, new generation is emerged by cross-over, mutation, and reproduction. The fitter individual has the higher probability to influence the latter generation of individuals. Therefore, by gradual improvement, the solutions move to the convergence toward the optimum. Finally, after reaching the termination criterion, the best found solution is reported.

No need for differentiability, convexity and continuity of the objective function is the major advantage of genetic algorithms. Furthermore, genetic algorithms can be relatively easily adjusted to almost every linear and non-linear problem. In order to reach the universal optimum, genetic algorithms with an elitist strategy have been verified, even in the case that the restrictions or objective function have non-smooth operators such as IF, MIN, MAX, and ABS functions. [37, 38].

The methodology of this paper is stated in main steps as follow:

1. Initialize a population of solutions randomly and
2. improve them by dominance properties (initialization).
3. Select parent chromosomes via the roulette wheel method (selection).
4. Do crossover process (crossover).
5. Implement mutation process (mutation).
6. Improve solutions by utilizing dominance properties (improvement).
7. Select survivors by using manner described in sub-section 5-2 (select survivors)
8. Repeat steps 2 to 6 till the termination criterion is met (termination).

The details of the proposed method are expressed in the remainder.

5. 1. Initialization

5. 1. 1. Solution Representation The solution representation offered in the first step to design an algorithm for the problem. The solution representation is a string of symbols containing solution characteristics setting up a bridge that connects the original problem space and the solution space being searched by the algorithm. The algorithm performance can be affected significantly by definition of an appropriate solution representation strategy.

Each solution of the considered problem is characterized by two sections including scheduling section and vehicle routing section. Deciding which job must be processed in-house and which job must be outsourced and determining the scheduling of the in-house jobs are encoded in scheduling section. Whether each customer is serviced by which vehicle or the rout of each vehicle is encoded in the vehicle routing section. The solution representation must map characteristics of the problem into array of numbers.

The solution representation is a matrix containing two parts with $N+f$ entries of real numbers which N denotes the number of all jobs and f denotes the number of customers. The first part characterizing the scheduling problem contains N entries of real numbers from interval [1, m+2) which m denotes the number of production lines. The second part characterizing the vehicle routing problem contains f entries of real numbers from interval [1, V+1) which V denotes the number of vehicles. A representation for a solution with 2 lines, 6 jobs, 4 customers and 2 vehicles is shown in Figure 1.

5. 1. 2. Decoding Procedure In this subsection, we propose a decoding procedure for genetic algorithm of the problem. Decoding procedure plays an important role in algorithm efficiency.

N

2.44	1.56	3.4	1.67	2.04	2.36

f

2.2	1.2	1.3	2.1

Figure 1. The solution representation

The jobs that should be produced with in-house lines and the jobs that should be outsourced as well as scheduling of the in-house jobs are indicated by decoding of the first part of the solution. The first part of the solution consists of real numbers having two integer and fractional parts. The integer part of each entry represents the number of the line producing the relative job, if the integer part is equal to 1,2,…,m then the job will be produced in-house; otherwise, if the integer part is equal to m+1 then the job will be outsourced. The job sequence for each in-house line is indicated from increasing order of the fractional parts of the relative entries.

The routes and the vehicles assigning to each route are indicated from the second part of the solution. The integer part of entry representing the vehicle number delivering the request of relative customer and delivery route for each vehicle is obtained from increasing order of the fractional parts of the relative entries. The solution represented in Figure 1 is decoded as follow: from scheduling part, jobs 2 and 4 are processed on line 1, jobs 5, 6 and 1 are processed on line 2 consecutively, and job 3 is outsourced; and from vehicle routing part, customers 2 and 3 are serviced by vehicle 1 and customers 2 and 4 by vehicle 2 consecutively.

5. 1. 3. Generation of the Initial Population

The initial population is generated by producing a population of individuals randomly, matrices with N+f entries of real numbers in determined intervals: N first entries from interval [1, m+2) and f remained entries from interval [1, V+1).

5. 2. Genetic Operators

In this section, the operators selection, reproduction, crossover, mutation as well as termination criteria used in proposed GA are presented as follow:

Selection: In GA search, selection strategy is considered as an important process leading to a proper performance direction. To select parents, one of the most effective selection strategies roulette wheel is used in this paper. The relative fitness can define whether an individual can be a parent according to fit_{ind}/Fit, where Fitdetermines the cumulated fitness of the current population. The fitness value of each individual is calculated from objective function of the problem in Equation (1).

Reproduction: Allowing the best organism(s) from the current generation to carry over to the next in the form of unchanged is regarded a practical variant of the general process of forming a new population. It supports the solution quality obtained by the GA and does not allow decreasing from one generation to the next.

Crossover: In order to produce offspring for the next generation, the chromosomes in the candidate population are subjected to crossover and mutation processes. In this article, the two-point crossover method is used with crossover probabilityp_c. In two-point crossover, two random points are picked from parent chromosomes to produce offsprings, and the bits between the two points are chosen according to the bits between the parents organisms.

Mutation: In this paper, the uniform mutation method is used with mutation probability p_m. In uniform mutation, the value of a randomly chosen gene is replaced with a uniform random value selected between [1, m+2) for production section of the chromosome and [1, V+1) for VRP section of the chromosome.

Survivor selection: survivor selection of the proposed algorithm is described in the following. A percentage of the fittest individuals of the current population (p_r) is carried over to the next population unaltered. If offspring obtained from crossover is fitter than relative parent, the offspring is copied to the next generation and the parents are removed. Offspring obtained from mutation operator is being moved to the next population. The remainders of the next population are provided by the fittest individuals in the current population. This survivor selection strategy guarantees avoidance of premature convergence and divergence. In the search space of the proposed GA, diversification provided by mutation and intensification is supplied by copying the fittest solutions on the next generation and local search provided by dominance properties.

Termination criteria: According to the order of met criteria, two termination criteria are proposed for the GADP:

- no improvement is seen for defined number of generation
- the time limit 600 s is elapsed.

6. COMPUTATIONAL STUDY

In this section, the efficiency of proposed algorithm is investigated. In the following, data are generated and parameters are initialized systematically. Afterwards, instances in varied size are generated and solved by proposed algorithm. Finally, the results are compared with those of MILP to analyse the efficiency of the algorithm.

6. 1. Data Generation

Based on literatures [8, 17], the production lines are considered identical and production time of one unit of each type of product$p_j = 1, j = 1,2,..,P$ is 1in each line. The setup times are chosen in a way that the instances satisfy the triangle inequality. Setup times are randomly chosen out of the interval [6, 10] with uniform distribution. In order to calculate setup cost, the amount of setup time is multiplied by value 25. There is no set-up time between producing jobs of the same type. All customers are randomly scattered in a square of locations from (0, 0) to

(100, 100). The processing site is located at point (50, 50). Between all pairs of customers and the processing site, the Euclidean distance is calculated. The variable transportation cost between destinations is equal to the travel times. The number of vehicles is equal to that of customers. The fixed cost for utilizing each vehicle is set to 250. The capacity of the vehicle is obtained from $\frac{3\sum_{ij} dem_{ij}}{f}$.

Based on literature [39], 70% of the demand of each job ij is randomly chosen out of the interval [40, 60] and the remaining is set to zero. The processing time of each job ij is computed through multiplying the demand by processing time of one unit of product j. The due dates are drawn from $[0.3\sum_{ij} p_{ij}, 0.7\sum_{ij} p_{ij}]$, the lead times from [100, 300] and the outsourcing cost from [100, 400] uniformly.

The rest of the parameters are chosen as follows. The rate of holding cost h_{ij} is randomly chosen out of the interval [1, 5]. The tardiness penalty cost is drawn randomly from interval [1, 10].

6. 2. Computational Results

Computational experiments are conducted to determine the performance of proposed mathematical model and GA hybridized with dominance properties (GADP). The results of proposed algorithm coded in C# are compared with those of MILP coded in GAMS and solved with CPLEX. All the computational experiments are performed using a computer with core i7 at 2.50GHz with12GB Ram.

Through a preliminary experiment, the GA parameters must be set empirically and the different values for each parameter should be evaluated. In order to determine the best value of these parameters, parameter tuning must be performed. The population size is examined on 15(N+f), 20(N+f), 30(N+f) and 50(N+f), crossover rate $p_c$has been tested on 0.75, 0.8 and 0.85, mutation rate $p_m$has been tested on 0.2, 0.15 and 0.1 and reproduction rate p_r has been tested on 0.05, 0.1 and 0.15. Each combination of the parameter values is tested on 6 randomly chosen instances to ensure that the GADP performance is robust. The obtained experimental results indicate that the following values can lead to satisfactory outcomes, population size: 20(N+f), p_c: 0.75, p_m: 0.1 and p_r: 0.1.

To analyse sensitiveness of key parameters, 7 randomly instances are chosen. Four and three different values for the parameters population size and crossover rate are tested respectively and three and three different values for the parameters mutation rate and reproduction rate are tested. Each test is run 10 times. The average cost and average computational time are shown in Table 1. It is obviously that increasing the population size has good impact in quality of results due to increase the search space. From Table 1, it is perceived that increasing in population size more than 20(N+f) leads to increasing in cost and computational time; it is occurred due to time limit (600 seconds).

In order to analyse the performance of proposed algorithm, generated instances are solved with some methods including:

- the MILP model implemented in GAMS in time limit 7200 seconds
- the proposed GA without the dominance properties
- the proposed GA hybridized with the dominance properties.

Now, the challenge in applying the proposed GA is that it may not perform better without the derived dominance properties; however, it may generate numerous additional solutions. Consequently, the performance comparison of the proposed GA with (GADP) and without (GA) the dominance properties can be used to prove their effectiveness and efficiency.

All instances including very small and small sizes as well as medium and large sizes are solved with the above stated methods and results are presented in Tables 2 and 3, respectively. As it is shown in the tables, the first column indicates the characteristics of the instances including number of the production lines (k), number of the products (P), number of the customers (f) as well as that of jobs (N). The values of k, P and f are imported as input data however number and configuration of the jobs (N) are randomly generated by the coded algorithm. In each row of the tables, minimum, average and maximum results obtained from each method for each instance are compared with the minimum solution achieved for that instance, besides, deviation percentage is mentioned as solution quality measure. In addition, the tables show the average CPU time, in seconds, for each problem instance spent by GAMS, GA, and GADP.

In Table 2, a value below MILP presented in boldface relates to a solution with optimality proven by GAMS, and an asterisk in Table 3 shows the failure of GAMS to identify even a possible solution within the imposed limited time. In Table 3, the times less than 7200 s spent by GAMS resulted in no optimal solution implies that GAMS has terminated due to memory limitations.

According to Tables 2 and 3, it can be concluded that obtaining optimal solutions for all of the instances with up to 2 production lines, 3 products, 5 customers and 14 jobs applying GAMS is possible. However, by increasing size of the instances, GAMS fails to find optimal solutions within a time limit of 7200 s; accordingly, for 2 instances with 56 and 57 jobs, it has failed to identify even a feasible solution within the imposed time limit.

In order to evaluate the significance of the differences among the results obtained by GA and GADP one-tailed paired t tests are performed. From Tables 2 and 3, a value below GADP presented in boldface relates to a solution with significant higher quality than solution obtained by GA. In 37 instances, results from GADP are significant better than those of GA.

TABLE 1. Sensitivity analyses of parameters

Population size (pop-size)		**Reproduction rate (p_r)**		**Mutation rate (p_m)**		**Crossover rate (p_c)**	
pop-size= 15(N+f)		p_r= 0.05		p_m= 0.05		p_c= 0.7	
average cost	average time	average cost	average time	average cost	average time	average cost	average time
27398	40	25839	59	26215	120	25318	62
pop-size= 20(N+f)		p_r= 0.1		p_m= 0.1		p_c= 0.8	
average cost	average time	average cost	average time	average cost	average time	average cost	average time
25318	70	25318	62	25318	62	26326	78
pop-size= 30(N+f)		p_r= 0.15		p_m= 0.2		p_c= 0.85	
average cost	average time	average cost	average time	average cost	average time	average cost	average time
25728	101	25632	81	26892	87	26893	93
pop-size= 50(N+f)							
average cost	average time						
26526	152						

TABLE 2. Very small and small instances

Instance		**MILP**		**GA**				**GADP**			
(k,P,f)	**N**	**Quality**	**Time**	**Min**	**Ave**	**Max**	**Time**	**Min**	**Ave**	**Max**	**Time**
	6	0	5	0	0	0	4	0	0	0	5
	7	0	11	0	0	0	7	0	0	0	8
(2,2,5)	8	0	80	0	0	0	9	0	0	0	6
	8	0	150	0	0	0	7	0	0	0	6
	9	0	162	0.01	0.01	0.01	11	0	0	0	14
Average		0	81.6	0	0	0	8	0	0	0	8
	10	0	1276	0	0	0	10	0	0	0	12
	10	0	1250	0	0	0	6	0	0	0	4
(2,2,7)	11	0	1327	0	0	0	6	0	0	0	4
	12	0	1537	0	0	0	27	0	0	0	19
	13	0	1522	0.03	0.03	0.03	18	0	0	0	16
Average		0	1443	0.01	0.01	0.01	13.4	0	0	0	11
	10	0	3622	0	0	0	15	0	0	0	17
	11	0	5502	0.01	0.03	0.05	12	0	0	0	10
(2,3,5)	11	0	2558	0	0	0	15	0	0	0	11
	12	0	2963	0	0	0	26	0	0	0	24
	12	0	1532	0.02	0.04	0.06	21	0	0.01	0.01	20
Average		0	3235.4	0.01	0.01	0.02	18	0	0	0	16
	15	0.03	2304	0.03	0.03	0.05	30	0	0	0	26
	16	0.05	2250	0.02	0.02	0.04	26	0	0	0	24
(2,3,7)	16	0.2	1902	0.03	0.06	0.12	29	0	0.03	0.04	13
	17	0.09	1509	0.01	0.03	0.06	21	0	0.01	0.02	14
	17	0.3	3264	0.01	0.04	0.08	30	0	0.02	0.05	42
Average		0.134	2242	0.02	0.04	0.07	27	0	0.01	0.02	24
	16	0.2	1282	0.02	0.04	0.06	26	0	0	0	25
	17	0.16	1302	0.01	0.03	0.05	21	0	0	0	11
(2,4,6)	17	0.19	1324	0.02	0.06	0.07	29	0	0	0	20
	18	0.4	2102	0.01	0.02	0.03	17	0	0	0.02	25
	18	0.39	2045	0.01	0.02	0.04	32	0	0.01	0.03	33

Average		0.268	1651	0.01	0.03	0.05	25	0	0	0.01	23
	18	0.2	1425	0	0.02	0.02	16	0	0	0	14
	18	0.24	1670	0.01	0.01	0.02	24	0	0	0	14
(2,5,5)	19	0.23	1710	0.04	0.04	0.04	25	0	0	0	15
	20	0.29	1754	0	0.01	0.02	16	0	0	0	18
	20	0.33	1826	0.01	0.01	0.02	18	0	0.01	0.02	16
Average		0.258	1677	0.01	0.02	0.03	20	0	0	0	15
Overall average		0.11	1721	0.01	0.02	0.03	22	0	0	0	16

TABLE 3. Medium and large instances

Instance		**MILP**		**GA**				**GADP**			
(k,P,f)	**N**	**Quality**	**Time**	**Min**	**Ave**	**Max**	**Time**	**Min**	**Ave**	**Max**	**Time**
	17	0.21	1357	0	0.01	0.02	25	0	0	0.01	15
	17	0.29	1331	0	0.01	0.03	16	0	0.01	0.03	18
(3,5,5)	18	0.22	1423	0	0.01	0.02	30	0	0.01	0.01	25
	18	0.25	1590	0	0	0.02	28	0	0	0.02	21
	20	0.2	1561	0	0.01	0.01	49	0	0.01	0.01	38
Average		0.23	1452	0	0.01	0.02	30	0	0.01	0.01	23
	24	0.31	1104	0	0.01	0.05	27	0	0	0.01	29
	24	0.35	1254	0.01	0.01	0.02	38	0	0	0	40
(3,5,7)	25	0.25	1038	0.01	0.01	0.03	37	0	0	0	28
	26	0.3	1361	0	0.02	0.04	61	0.02	0.03	0.04	33
	26	0.26	3995	0.02	0.03	0.03	43	0	0.01	0.01	40
Average		0.29	1750	0.01	0.02	0.03	41	0	0.01	0.01	34
	28	0.3	1602	0	0.02	0.08	72	0	0	0	80
	29	0.39	1632	0	0.01	0.03	65	0	0.01	0.01	93
(3,4,10)	30	0.2	1423	0	0.05	0.11	55	0	0.04	0.08	65
	30	0.35	1506	0	0.06	0.14	55	0	0.02	0.03	53
	31	0.37	1610	0	0.06	0.06	160	0	0.02	0.06	65
Average		0.23	1554	0	0.04	0.08	82	0	0.02	0.04	71
	38	0.23	2010	0	0.04	0.08	137	0	0.01	0.03	115
	39	0.28	7200	0	0.03	0.08	165	0	0.03	0.05	144
(3,5,11)	40	0.23	4251	0.04	0.05	0.08	155	0	0.01	0.01	133
	40	0.27	7200	0.02	0.07	0.11	140	0	0.01	0.03	146
	41	0.15	1568	0.02	0.08	0.12	150	0	0.02	0.04	142
Average		0.23	4390	0.02	0.05	0.1	150	0	0.02	0.09	136
	45	0.38	5487	0.05	0.09	0.12	224	0	0.02	0.03	223
	45	0.4	7200	0.08	0.13	0.16	189	0	0.08	0.12	196
(3,5,13)	45	0.1	3546	0.03	0.06	0.08	217	0	0.04	0.08	199
	47	0.27	6754	0	0.01	0.02	240	0	0.01	0.03	229
	48	0.3	6542	0.1	0.14	0.16	246	0	0.06	0.12	236
Average		0.29	5890	0.05	0.09	0.11	223	0	0.04	0.08	216
	54	0.35	7200	0.02	0.08	0.1	354	0	0.02	0.05	351
	55	0.14	6723	0.01	0.04	0.08	362	0	0.02	0.04	357
(4,7,10)	56	0.32	7200	0.05	0.09	0.11	384	0	0.02	0.03	374
	57	*	7200	0.02	0.04	0.06	402	0	0.03	0.04	396
	57	*	7200	0	0.03	0.06	431	0	0.04	0.07	420
Average		0.29	7104	0.02	0.06	0.08	387	0	0.03	0.05	380
Overall average		0.26	3690	0.02	0.05	0.07	153	0	0.02	0.05	143

GA can obtain optimal solutions for 10 small instances. On the very small and small instances and some of the medium instances, the results obtained from GA are comparable with those of MILP in the solution quality, although GA outperforms MILP in terms of CPU time; by increasing the size of the instances, GA outperforms in terms of solution quality as well.

Furthermore, GADP is able to achieve optimal solutions for all of the instances including up to 14 jobs that the optimality is verified by GAMS. GADP is successful in performance even for the instances with a large number of jobs; in addition, the worst and the best results obtained by GADP indicate insignificant differences.

Although GADP and GA both perform the same in terms of solution quality for the some instances with up to 2 production lines, 3 products, 5 customers and 12 jobs, by increasing the size of instances, GA fails to compete with GADP; even the worst results achieved by GADP are better in comparison with the best results obtained by GA.

Moreover, according to CPU time, except for four small instances, GA requires more computational effort than GA, due to lack of the dominance properties which play a significant role in the intensification of the search; by producing numerous additional solutions compared to GADP, GA converges to a final solution.

Two interval plots are depicted for final solution (cost) and computational time. Error ratio of these interval plots is set 0.05. The 95% confidence interval plots of the three algorithms have been depicted for the cost in Figure 2. The 95% confidence interval plots of these algorithms are shown for the computational time in Figure 3. From Figure 2, it is perceived that the two algorithms have significant differences in obtained final solution. Moreover, the considerable differences among the computational time of GA and GADP.

Therefore, applying the dominance properties to obtain better solutions in comparatively shorter computation times is recommended. Generally, based on the computational results, GADP can be regarded as fast and robust as well as capable to provide an efficient approach to gain optimal or near- optimal solutions with small computational necessity.

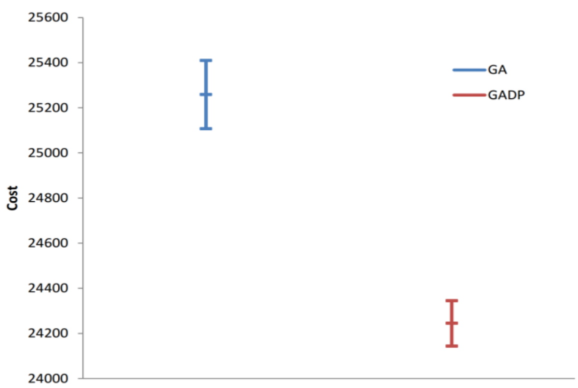

Figure 2. Confidence interval of GA and GADP for final solution

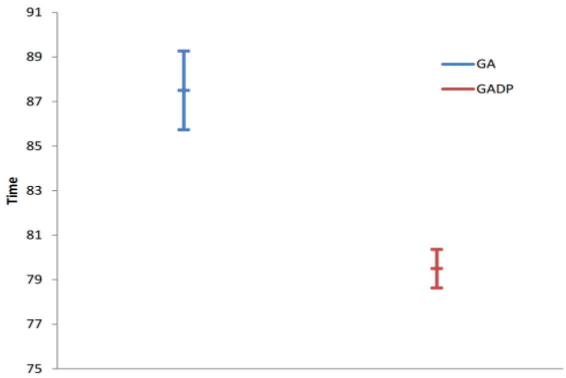

Figure 3. Confidence interval of GA and GADP for computational time

7. CONCLUSION

The need to minimize the amount of inventory across the supply chain and to be responsive to customers' requests indicates the importance of using cooperated production and distribution models. To handle demand fluctuations without holding a high inventory forces manufacturer to outsource some jobs to a sub-contractor.

In this paper, a new integrated production scheduling, vehicle routing, inventory and outsourcing problem is modelled. Production phase considers parallel machine scheduling including setup times with outsourcing allowed and distribution phase investigates batch delivery by a fleet of homogenous vehicles with respect to holding cost of completed jobs. An example of this problem can be seen in dairy factory and paste factory. The aim of this paper is to find a schedule for joint in-house production and distribution as well as to determine the jobs that must be outsourced in the way that minimizes total cost of production, holding, outsourcing and distribution. The production cost consists of sequence-dependent set-up cost. The distribution cost consists of tardiness penalties and fixed vehicle usage cost and variable travel time proportional cost.

This paper presents a mathematical model for describing the problem and designs a hybrid algorithm using dominance properties combined with Genetic Algorithm (GA). Generating the initial population and improving the quality of solutions generated by the GA can be significantly affected by the dominance properties applied as a local search strategy. Finally, computational experiments are performed to evaluate the performance of solution approach. Therefore, applying the dominance properties to obtain better solutions in comparatively shorter computation times is recommended. Generally, based on the computational results, GADP can be regarded as fast and robust as well as being capable to provide an efficient approach to gain optimal or near-optimal solutions with small computational necessity.

For future studies, we suggest utilizing a two-level distribution model including distribution centres in the first level and customers (i.e. retailers) in the second level. This type of distribution is able to cover more customer areas in scattered geographic location. Furthermore, outsourcing option is rarely considered in the literature. It is suggested to study the context of outsourcing option in the future works to fill this research gap.

8. REFERENCES

1. Taghizadehalvandi, M., and Kamisli Ozturk, Z. "Multi-objective Solution Approaches for Employee Shift Scheduling Problems in Service Sectors (RESEARCH NOTE)." ***International Journal of Engineering, Transactions C: Aspects*** , Vol. 32, No. 9, (2019), 1312–1319. https://doi.org/10.5829/ije.2019.32.09c.12
2. Amin-Tahmasb, H., and Hami, M. "Optimization of a Bi-objective Scheduling for Two Groups of Experienced and Inexperienced Distribution Staff Based on Capillary Marketing." ***International Journal of Engineering, Transactions B: Applications***, Vol. 31, No. 11, (2018), 1943–1952. https://doi.org/10.5829/ije.2018.31.11b.19
3. Sarmiento, A. M., and Nagi, R. "A review of integrated analysis of production-distribution systems." ***IIE Transactions (Institute of Industrial Engineers)***, Vol. 31, No. 11, (1999), 1061–1074. https://doi.org/10.1080/07408179908969907
4. Chang, Y. C., Li, V. C., and Chiang, C. J. "An ant colony optimization heuristic for an integrated production and distribution scheduling problem." ***Engineering Optimization***, Vol. 46, No. 4, (2014), 503–520. https://doi.org/10.1080/0305215X.2013.786062
5. Ahmadizar, F., and Amiri, Z. "Outsourcing and scheduling for a two-machine flow shop with release times." ***Engineering Optimization***, Vol. 50, No. 3, (2018), 483–498. https://doi.org/10.1080/0305215X.2017.1325483
6. McIvor, R. "The influence of capability considerations on the outsourcing decision: The case of a manufacturing company." ***International Journal of Production Research***, Vol. 48, No. 17, (2010), 5031–5052. https://doi.org/10.1080/00207540903049423
7. Qi, X. "Outsourcing and production scheduling for a two-stage flow shop." ***International Journal of Production Economics***, Vol. 129, No. 1, (2011), 43–50. https://doi.org/10.1016/j.ijpe.2010.08.011
8. Amorim, P., Belo-Filho, M. A. F., Toledo, F. M. B., Almeder, C., and Almada-Lobo, B. "Lot sizing versus batching in the production and distribution planning of perishable goods." ***International Journal of Production Economics***, Vol. 146, No. 1, (2013), 208–218. https://doi.org/10.1016/j.ijpe.2013.07.001
9. Van Buer, M. G., Woodruff, D. L., and Olson, R. T. "Solving the medium newspaper production/distribution problem." ***European Journal of Operational Research***, Vol. 115, No. 2, (1999), 237–253. https://doi.org/10.1016/S0377-2217(98)00300-2
10. Chen, Z. L., and Vairaktarakis, G. L. "Integrated scheduling of production and distribution operations." ***Management Science***, Vol. 51, No. 4, (2005), 614–628. https://doi.org/10.1287/mnsc.1040.0325
11. Li, C. L., and Vairaktarakis, G. "Coordinating production and distribution of jobs with bundling operations." ***IIE Transactions (Institute of Industrial Engineers)***, Vol. 39, No. 2, (2007), 203–215. https://doi.org/10.1080/07408170600735561
12. Chang, Y. C., and Lee, C. Y. "Machine scheduling with job delivery coordination." ***European Journal of Operational Research***, Vol. 158, No. 2, (2004), 470–487. https://doi.org/10.1016/S0377-2217(03)00364-3
13. Li, C. L., Vairaktarakis, G., and Lee, C. Y. "Machine scheduling with deliveries to multiple customer locations." ***European Journal of Operational Research***, Vol. 164, No. 1, (2005), 39–51. https://doi.org/10.1016/j.ejor.2003.11.022
14. Geismar, J. H. N., Laporte, G., Lei, L., and Sriskandarajah, C. "The integrated production and transportation scheduling problem for a product with a short lifespan." ***INFORMS Journal on Computing***, Vol. 20, No. 1, (2008), 21–33. https://doi.org/10.1287/ijoc.1060.0208
15. Chen, H. K., Hsueh, C. F., and Chang, M. S. "Production scheduling and vehicle routing with time windows for perishable food products." ***Computers and Operations Research***, Vol. 36, No. 7, (2009), 2311–2319. https://doi.org/10.1016/j.cor.2008.09.010
16. Ullrich, C. A. "Integrated machine scheduling and vehicle routing with time windows." ***European Journal of Operational Research***, Vol. 227, No. 1, (2013), 152–165. https://doi.org/10.1016/j.ejor.2012.11.049
17. Belo-Filho, M. A. F., Amorim, P., and Almada-Lobo, B. "An adaptive large neighbourhood search for the operational integrated production and distribution problem of perishable products." ***International Journal of Production Research***, Vol. 53, No. 20, (2015), 6040–6058. https://doi.org/10.1080/00207543.2015.1010744
18. Kang, H. Y., Pearn, W. L., Chung, I. P., and Lee, A. H. I. "An enhanced model for the integrated production and transportation problem in a multiple vehicles environment." ***Soft Computing***, Vol. 20, No. 4, (2016), 1415–1435. https://doi.org/10.1007/s00500-015-1595-7
19. Li, K., Zhou, C., Leung, J. Y. T., and Ma, Y. "Integrated production and delivery with single machine and multiple vehicles." ***Expert Systems with Applications***, Vol. 57, (2016), 12–20. https://doi.org/10.1016/j.eswa.2016.02.033
20. Karaoğlan, İ., and Kesen, S. E. "The coordinated production and transportation scheduling problem with a time-sensitive product: a branch-and-cut algorithm." ***International Journal of Production Research***, Vol. 55, No. 2, (2017), 536–557. https://doi.org/10.1080/00207543.2016.1213916
21. Devapriya, P., Ferrell, W., and Geismar, N. "Integrated production and distribution scheduling with a perishable product." ***European Journal of Operational Research***, Vol. 259, No. 3, (2017), 906–916. https://doi.org/10.1016/j.ejor.2016.09.019
22. Lacomme, P., Moukrim, A., Quilliot, A., and Vinot, M. "Supply chain optimisation with both production and transportation integration: multiple vehicles for a single perishable product." ***International Journal of Production Research***, Vol. 56, No. 12, (2018), 4313–4336. https://doi.org/10.1080/00207543.2018.1431416
23. Tamannaei, M., and Rasti-Barzoki, M. "Mathematical programming and solution approaches for minimizing tardiness and transportation costs in the supply chain scheduling problem." ***Computers and Industrial Engineering***, Vol. 127, (2019), 643–656. https://doi.org/10.1016/j.cie.2018.11.003
24. Tavares-Neto, R. F., and Nagano, M. S. "An Iterated Greedy approach to integrate production by multiple parallel machines and distribution by a single capacitated vehicle." ***Swarm and Evolutionary Computation***, Vol. 44, (2019), 612–621. https://doi.org/10.1016/j.swevo.2018.08.001

25. Mohammadi, S., Al-e-Hashem, S. M. J. M., and Rekik, Y. "An integrated production scheduling and delivery route planning with multi-purpose machines: A case study from a furniture manufacturing company." ***International Journal of Production Economics***, Vol. 219, (2020), 347–359. https://doi.org/10.1016/j.ijpe.2019.05.017
26. Mousavi, M., Hajiaghaei–Keshteli, M., and Tavakkoli–Moghaddam, R. "Two calibrated meta-heuristics to solve an integrated scheduling problem of production and air transportation with the interval due date." ***Soft Computing***, Vol. 24, No. 21, (2020), 16383–16411. https://doi.org/10.1007/s00500-020-04948-y
27. L., I., F., A., and J., A. "A Hybrid Imperialist Competitive Algorithm for Integrated Scheduling of Production and Distribution with Vehicle Routing." ***ournal of Industrial Engineering Research in Production Systems***, Vol. 6, No. 12, (2018), 63–81. Retrieved from https://www.sid.ir/en/Journal/ViewPaper.aspx?ID=711650
28. Ahmadizar, F., and Farhadi, S. "Single-machine batch delivery scheduling with job release dates, due windows and earliness, tardiness, holding and delivery costs." ***Computers and Operations Research***, Vol. 53, (2015), 194–205. https://doi.org/10.1016/j.cor.2014.08.012
29. Chang, P. C., Chen, S. H., and Mani, V. "A hybrid genetic algorithm with dominance properties for single machine scheduling with dependent penalties." ***Applied Mathematical Modelling***, Vol. 33, No. 1, (2009), 579–596. https://doi.org/10.1016/j.apm.2008.01.006
30. Chang, P. C., and Chen, S. H. "Integrating dominance properties with genetic algorithms for parallel machine scheduling problems with setup times." ***Applied Soft Computing Journal***, Vol. 11, No. 1, (2011), 1263–1274. https://doi.org/10.1016/j.asoc.2010.03.003
31. Ahmadizar, F., and Hosseini, L. "Bi-criteria single machine scheduling with a time-dependent learning effect and release times." ***Applied Mathematical Modelling***, Vol. 36, No. 12, (2012), 6203–6214. https://doi.org/10.1016/j.apm.2012.02.002
32. Holland, J. H. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control and artificial intelligence. MIT press, 1975. Retrieved from https://ui.adsabs.harvard.edu/abs/1975anas.book.....H/abstract
33. Goldenberg, D. Genetic algorithms in search, optimization and machine learning. Addison-Wesley Publishing Company, 1989.
34. Davis, L. Handbook of genetic algorithms. Van Nostrand Reynold, 1991. Retrieved from http://papers.cumincad.org/cgi-bin/works/paper/eaca
35. Talbi, E. Metaheuristics: from design to implementation. John Wiley & Sons, 2009. Retrieved from https://books.google.com/books?hl=en&lr=&id=SIsa6zi5XV8C&oi=fnd&pg=PR7&ots=-bPJrReuCr&sig=kqHIxfQssPePngA0_zVAfx3ljQs
36. Gendreau, M., and Potvin, J. Handbook of metaheuristics. Springer, 2010. Retrieved from https://link.springer.com/content/pdf/10.1007/978-1-4419-1665-5.pdf
37. Suzuki, J. "A Markov Chain Analysis on Simple Genetic Algorithms." ***IEEE Transactions on Systems, Man, and Cybernetics***, Vol. 25, No. 4, (1995), 655–659. https://doi.org/10.1109/21.370197
38. Lozano, J. A., Larrañaga, P., Graña, M., and Albizuri, F. X. "Genetic algorithms: Bridging the convergence gap." ***Theoretical Computer Science***, Vol. 229, No. 1–2, (1999), 11–22. https://doi.org/10.1016/S0304-3975(99)00090-0
39. Tavares Neto, R. F., Godinho Filho, M., and da Silva, F. M. "An ant colony optimization approach for the parallel machine scheduling problem with outsourcing allowed." ***Journal of Intelligent Manufacturing***, Vol. 26, No. 3, (2015), 527–538. https://doi.org/10.1007/s10845-013-0811-5

Persian Abstract

چکیده

در این مقاله، یک مسأله جدید زمانبندی ادغامی تولید و مسیریابی وسایل نقلیه، با در نظر گرفتن برونسپاری و موجودی مورد مطالعه قرار گرفته است. بخش تولید، زمانبندی ماشین‌های موازی با در نظر گرفتن زمان‌های آماده‌سازی و مجاز بودن برونسپاری را بررسی می‌کند و بخش توزیع، تحویل دسته‌ای توسط یک ناوگان از وسایل حمل یکسان با در نظر گرفتن هزینه نگه‌داری کارهای تکمیل شده را بررسی می‌کند. مسأله به صورت برنامه‌ریزی خطی عدد صحیح فرموله شده و هدف، حداقل ساختن مجموع هزینه‌های تولید، برونسپاری، نگه‌داری، تأخیر و ثابت و متغیر حمل است. به دلیل Np-hard بودن، برای حل مسأله تعدادی قواعد غلبه استخراج شده و با یک الگوریتم ژنتیک ادغام شده است. برای ارزیابی کارایی و اثربخشی الگوریتم ترکیبی پیشنهادی، بر روی تعدادی نمونه مسائل تصادفی تولیدی مطالعه عددی انجام شده است. به منظور نشان دادن تأثیر پارامترهای کلیدی بر تابع هدف، آنالیز حساسیت انجام شده است. برای ارزیابی معناداری تفاوت جواب‌های به دست آمده از الگوریتم پیشنهادی با الگوریتم ژنتیک، آزمون آماری t انجام شده و نمودارهای بازه‌ای رسم شده است.