

Integrated locating in-house logistics areas and transport vehicles selection problem in assembly lines

Amir Nourmohammadi^{a,b}, Hamidreza Eskandari^{c,d,*}, Masood Fathi^a and Amos H.C. Ng^a

^a*Department of Production and Automation Engineering, University of Skövde, P.O. Box 408, SE-541 28, Skövde, Sweden*

^b*Faculty of Industrial & Systems Engineering, Tarbiat Modares University, P.O. Box 14115-138, Tehran, Iran*

^c*Faculty of Management and Economics, Tarbiat Modares University, Tehran, Iran*

^d*School of Management, Swansea University, Swansea, UK*

*corresponding author: Email: eskandari37@gmail.com; Telephone: (+98) 21 22291235

Abstract: Decentralized in-house logistics areas, known as supermarkets, are widely used in the manufacturing industry for parts feeding to assembly lines. In contrary to the literature and inspired by observation in a real case, this study relaxes the assumption of using identical transport vehicles when deciding on the supermarkets' location by considering the availability of different vehicles. In this regard, this study deals with the integrated supermarket location and transport vehicles selection problems (SLTVSP). A mixed-integer programming (MIP) model of the problem is developed. Due to the complexity of the problem, a hybrid genetic algorithm (GA) with variable neighborhood search (GA-VNS) is also proposed to address large-sized problems. The performance of GA-VNS is compared against the MIP, the basic GA, and simulated annealing (SA) algorithm. The computational results from the real case and a set of generated test problems show that GA-VNS provides a very good approximation of the MIP solutions at a much shorter computational time while outperforming the other compared algorithms. The analysis of the results reveals that it is beneficial to apply different transport vehicles rather than identical vehicles for SLTVSP.

Keywords: In-house logistics; supermarket location; parts feeding; transport vehicles; mixed-integer programming; genetic algorithm.

1. Introduction

During the past years, there has been a shift from process improvement of the assembly lines to the part logistics operations. To satisfy the consumers' needs, manufacturers have to customize their products from an assortment of options resulting in different product models to be produced and a large variety of parts to be fed to the stations. Furthermore, according to the just-in-time (JIT) concept, the delivery cycles have to be reduced while the need for well-organized and consistent logistics operations has increased (Battini et al. 2013). In line with these shifts, most manufacturers, particularly in the automotive industry, have implemented the so-called supermarkets to overcome the above challenges faced by today's demanding markets while following the just-in-time (JIT) concept. Supermarkets are decentralized in-house

logistics storage areas that are scattered over the assembly line (AL) and near to the stations (Boysen et al., 2015). Supermarkets are generally supplied by the central storage areas with large-sized trucks, while the AL stations are fed by smaller transport vehicles. Using this policy, the parts feeding can be managed in a fast, flexible and reliable manner so that responding to unforeseen events can be quickly re-planned by regular small-lot deliveries over short distances. Moreover, the inventory level of the stations is reduced while replenishments from the central storage which is usually located in a far distance from the AL are avoided. In addition, small-lots entail the replacement of the parts in smaller bins which can be stored and accessed more ergonomically and efficiently near the AL so that the pressure on the operators is diminished and the handling time is reduced (Battini et al. 2013). The complete description of the part supply process using supermarkets can be found in Emde and Boysen (2011, 2012) and Battini et al. (2013).

Despite the above-mentioned advantages, the planning and implementing of supermarkets requires addressing several interrelated decision problems (Emde and Boysen, 2012) known as supermarket-related decision problems, including: (i) decide on the number and location of supermarkets, namely, the supermarket location problem (SLP); (ii) decide on the number of tow trains and the stations that each serve as a routing problem (RP); (iii) decide on the fixed delivery schedule of each transport vehicle on its associated route as a scheduling problem (SP); and finally, (iv) decide on the number and types of part bins to be loaded per tour of tow trains as a loading problem (LP).

In spite of the importance of the first decision problem (i.e., SLP) as the most long-term problem on the shop floor where the space is highly valuable, very few studies have considered its effect on other operational decision problems mentioned above (Battini et al. 2013). Thus, this study aims to address the SLP by taking into account more real-world considerations. Unlike many related studies in the literature that have assumed only identical transport vehicles (i.e., tow train) are used to deliver bins of parts to the stations, this study considers that other types of transport vehicles (e.g., automatic guided vehicles (AGVs) and trollies) with different bin sizes can also be applied for parts delivery to the stations. The use of multi-type transport vehicles has been inspired by consulting some logistics experts and observation of real-world industrial practices in an automotive assembly plant. Based on the observed practices in the real-world industry, it can be argued that it is more beneficial to apply different transport vehicles than can be fully loaded based on the demand of the associated parts. It is worth mentioning that applying different transport vehicles in just-in-time supermarkets can be widely employed in lean manufacturing of all different types of assembly lines, including automotive (Emde and Gendreau, 2017) and trucks or busses (Emde et al., 2018), to name a few. For instance, when a small number of bins have to be supplied to a sequence of stations, they prefer to send the required parts by trolley rather than tow train to ensure that the capacity of the transport vehicle is efficiently used. On the other hand, when a large number of bins have to be supplied to the stations, transportation using a tow train might be more efficient/beneficial than applying several AGVs/trollies. The managers at the company want to decide about the most appropriate mode of transportation from each established supermarket. By assuming that the decisions on determining the type and the number of transport vehicles per supermarket can be defined as a new decision problem called transport vehicles selection problem (TVSP),

this study aims to deal with the integrated supermarket location and transport vehicles selection problem, so-called SLTVSP in brief. The integrated problem are first formulated as a mixed integer programming (MIP) model.

Furthermore, due to the inherent complexity of basic SLP (Alnahhal and Noche, 2015; Zhou and Tan, 2019), it becomes even more complex when it is integrated with the TVSP in this study so that resorting to meta-heuristics is the main option especially if the problem size is large. On the other hand, since genetic algorithms (GAs) have shown a good performance in addressing a large variety of optimization problems, such as scheduling (Zhao et al., 2017; Zou et al., 2018), logistics and supply chain (Dolgui et al., 2018), assembly line balancing (Belassiria et al., 2018), and SLP (Alnahhal and Noche, 2015), in this paper a GA with new, customized mechanisms is proposed to efficiently solve the considered problems. The proposed GA can determine the exact places where the supermarkets are located while addressing the SLP. Moreover, to be able to address the integrated SLP and TVSP, a new representation is proposed so that the assignments of stations to the supermarkets along with the assignment of transport vehicles to the supermarkets are performed simultaneously. Considering the above GA characteristics, new initialization, crossover and mutation operators are also developed for the proposed GA. Moreover, to improve the local search capability of the proposed GA, it is hybridized with a variable neighborhood search (VNS).

To sum up, the main contributions of this study are as follows:

- (1) For the first time in the literature, this study simultaneously deals with two main in-house logistics decision problems, namely SLP and TVSP, while considering the use of different types of transport vehicles.
- (2) A new MIP model for the integrated SLP and TVSP (SLTVSP) is proposed where the total cost of parts feeding (PF) including the supermarket installation, transport vehicles' procurement, and parts shipment costs, is minimized.
- (3) A GA-VNS hybridized algorithm is developed by introducing a customized solution representation and initialization mechanisms. The GA-VNS also benefits from new crossover and mutation operators that are tailored to the problem representation ensuring the generation of feasible solutions.

The remainder of this paper is structured as follows. Section 2 discusses the scope of the study and provides a recent literature review on supermarket-related decision problems. An explanation of the real case study in an automotive component assembly plant and the mathematical model of the considered problem are presented in Section 3. The proposed GA-VNS is presented in Section 4. The computational study is performed in Section 5. Finally, the concluding remarks are outlined in Section 6.

2. Scope of study and literature review

Considering today's industrial paradigm, many companies/organizations have attempted to move towards sustainability throughout their production and supply chain systems. To be able to accord with these ongoing trends, companies have to transfer from their traditional linear business/economy model of "take, make, use and disposal" in an open-end production system

towards a circular business model that entails closed-loop production and supply chain networks (Geissdoerfer et al., 2018). The above transformation requires the companies to implement an efficient use/restore of their resources throughout their products' life cycle. To cope with the above challenges, companies are advised to take two main measures to reinforce the circularity between the two extreme points of their business models (Urbinati et al., 2017): (1) by proposing value towards the customer ;(2) by proposing value towards their supply chain network. The former can be achieved by the implementation of Industry 4.0 concept that promotes the products' features to be directly designed by the customers, which is made possible using digital manufacturing and Internet of Things technologies in smart factories (Lopes de Sousa Jabbour et al., 2018). The latter includes both the outbound logistics, e.g., through interaction with their suppliers realized as inter-organizational cooperation and the inbound logistics, e.g., through reorganizing their internal material feeding strategies realized as intra-organizational process. In this regard, cross-dock is perceived as inter-organizational cooperation in the supply chain network to consolidate smaller shipments of several companies to full truckloads realized as economies in outbound logistics (Boysen and Fliedner, 2010). On the contrary, supermarket, known as the equivalent of cross-dock within inbound logistics, has been recently applied by many manufacturers to improve the intra-organizational logistics activities. It is worth mentioning that the scope of this study relies on the next important decision problem after the supermarket is applied within inbound logistics as approached by Emde and Boysen (2012) and Zhou and Tan (2019). Readers who are interested in knowing more about the state-of-the-art economy business models in supply chain management and logistics are referred to van Buren et al. (2016) and Geissdoerfer et al. (2018).

Since applying supermarket is a relatively new concept, there are only a few studies dealing with supermarket-related decision problems. Emde and Boysen (2011) dealt with the routing problem (ii) and scheduling problem (iii), simultaneously. A dynamic programming approach with polynomial-time was proposed to optimize the inventory cost and the equipment cost in terms of the number of tow trains with a limited capacity. Emde et al. (2012) addressed the loading problem (iv) of tow trains, given a predefined route and schedule. Given the capacity of tow trains, an exact solution approach was proposed to optimize the inventory cost associated with the station demand while no shortages were allowed. Golz et al. (2012) coped with the routing (ii), scheduling (iii), and loading (iv) problems, simultaneously. They proposed a heuristic method to optimize the number of tow trains with limited capacity given a set of predetermined routes while no shortages were allowed. Fathi, Alvarez, et al., (2014), Fathi, Rodríguez, et al., (2014, 2016) dealt with tow train scheduling (iii) and loading (iv) problems, simultaneously. In the first two studies, the integrated problem was formulated as a mixed-integer programming model, and a simulated annealing and a memetic ant colony optimization algorithm were proposed to optimize the number of tours and the inventory level. In the third study, the authors incorporated the tour time and weight constraints into their previous model and proposed a modified particle swarm optimization algorithm to solve it. Emde and Gendreau (2017) considered the scheduling (iii) problem of tow trains used in the delivery of part from supermarkets with limited capacities. They proved that the problem is complex (NP-hard) in nature. Thus, exact and heuristic approaches were proposed to address the real industrial instances by minimizing the in-process inventory cost. Emde et al. (2018) addressed the

scheduling (iii) problem of electric vehicles used in part feeding in an engine manufacturing company in Germany. Given a specific time interval between the vehicles round trips to stations, they proposed a heuristic algorithm to optimize the number of vehicles considering the electricity recharging of the vehicles. The above studies have dealt with more operational decision problems. As for the focus of this study, less than a handful of studies have addressed the SLP (i) as the strategic decision problem of PF using supermarket. The most relevant studies to the current article are reported below. The interested readers are referred to Boysen et al. (2015) and Schmid and Limère (2019) for further studies in the part logistics and their related decision problems.

Battini et al. (2010) proposed a step-by-step decision support framework to determine the degree of centralization/decentralization for each part supply and also the place where the supermarket can be established, considering transportation, inventory and installation costs. They addressed the SLP (i) by assuming that a single supermarket can be implemented to supply several ALs. This is while many real-world manufacturers have been employing more than a few supermarkets to feed each of their ALs. Emde and Boysen (2012) proposed a dynamic programming (DP) approach for SLP (i). They applied DP to minimize the transportation and the installation costs of supermarkets. The authors hypothesized that the placement of supermarkets could be determined in any place around the stations. However, since space is scarce and valuable on the shop floor, this assumption can result in impractical locations for supermarkets. Additionally, there might be particular places on the shop floor that cannot be employed by supermarkets as they are occupied by other facilities. Alnahhal and Noche (2015) proposed a simple MIP model for SLP (i) in which the total shipment and installation costs of supermarkets were minimized while the capacity of the supermarkets, as well as the possible places for supermarket placements, were taken into consideration. They also proposed a GA and compared its performance with the MIP model. However, their GA does not provide any information about the supermarket number feeding the stations. Nourmohammadi et al. (2019) dealt with the decision problem of configuring a new AL with stochastic task times and demands in which mathematical programming was proposed to address both ALBP and SLP (i) in a hierarchical approach. The objectives of the proposed SLP model were to optimize the inventory, shipment, and installation costs of supermarkets. Recently, Zhou and Tan (2019) addressed the SLP (i) by considering the limited capacity and the utilization rate of the supermarkets. They proposed an algorithm based on differential evolution to optimize the installation/operating cost of the supermarket and the transportation cost of tow trains.

In all the above studies, it is assumed that an identical transport vehicle (tow train) is available for PF using supermarkets. However, there might be different transport vehicles available for PF in the real-world (as observed in the real case of this study) such as AGV, trolley or conveyor to be chosen from due to transportation system or labor costs, space limitation and other considerations. To the best of the authors' knowledge, only Battini et al. (2015) have considered the transport vehicles selection problem (TVSP), particularly within the in-plant milk runs systems. However, they proposed a decision support framework based

on a multi-scenario analysis tool to suggest which transportation mode should be chosen for PF. Moreover, they did not consider TVSP while addressing the SLP.

Table 1 shows a summary of the supermarket decision problems reviewed in this study, including the problem characteristics, objectives, and solution approaches.

Table 1. Summary of the supermarket decision problems reviewed.

Study	Decision problem				Problem characteristics				Objectives				Solution approach					
	SL	RP	SP	LP	TVSP	CSP	LC	RC	DTV	IC	SC	INC	EC	DSF	MP	HA	MA	EA
Battini et al. (2010)	•					•		•		•	•	•		•				
Emde and Boysen (2011)		•	•				•					•	•		•			•
Emde et al. (2012)				•			•					•			•			•
Golz et al. (2012)	•	•	•				•	•					•			•		
Emde and Boysen (2012)	•							•		•	•				•			•
Fathi, Alvarez, et al. (2014)			•	•			•	•			•*	•			•			•
Fathi, Rodríguez, et al. (2014)			•	•			•	•			•*	•			•			•
Alnahhal and Noche (2015)	•					•	•			•	•				•			•
Battini et al. (2015)					•			•	•		•		•	•				
Fathi et al. (2016)			•	•			•	•			•*	•			•			•
Emde and Gendreau (2017)			•				•	•				•			•	•		•
Emde et al. (2018)			•				•	•					•		•			
Nourmohammadi et al. (2019)	•					•	•	•		•	•	•			•			•
Zhou and Tan (2019)	•					•	•			•	•				•			•
This study	•				•	•	•	•	•	•	•	•	•	•	•			•

Note: Candidate Supermarket Places (CSP), Limited Capacity (LC) of supermarket/tow train, Real Case (RC), Different Transport Vehicles (DTV), Installation Cost (IC), Shipment Cost (SC), Inventory Cost (INC), Equipment Cost (EC), Decision Support Framework (DSF), Mathematical Programming (MP), Heuristic Algorithm (HA), Meta-heuristic Algorithm (MA), Exact Approach (EA). *The shipment cost is in terms of the number of tours taken.

According to Table 1, no study in the literature has yet dealt with both SLP and TVSP, simultaneously. It is worth mentioning that any decision regarding TVSP is highly dependent upon SLP and vice versa. Moreover, in the existing literature, it is assumed that the capacity of transport vehicles and accordingly their shipment costs are equal without taking the mode of transportation into consideration. In spite of different studies that have contributed to the supermarket-related literature so far, there is still a wide gap between the theory and application. Thus, this study which is inspired by the observation made in a real-world case aims to alleviate this gap by relaxing the assumption that different transport vehicles can be used at supermarkets to feed part bins to the stations.

3. Problem description and formulation

In this section, the problem considered in this study is described in detail, and the proposed model to deal with the described problem is presented.

3.1. Problem description

This study is inspired by the observation made in a real-world AL at an automotive manufacturing company in which two different part feeding policies were used. In the line stocking policy, the parts which require no further processing are directly shipped from the central warehouse to be stored next to the stations in large/small containers, depending on the packaging received from the suppliers. In the downsizing policy, first, the parts are delivered in large containers to the decentralized warehouses (supermarkets), where parts are repacked into bins of standard size. Then, from supermarkets, the bins can be loaded on particular transport vehicles to be regularly supplied to the stations following a predetermined production plan. Regarding the second part feeding policy for parts with high demand rates, the decision-makers (DMs) at the company aim to establish some new supermarkets close to the AL so that the parts can be frequently delivered to the stations in small bins. Thus, first, the DMs would like to determine the optimal number and the location of supermarkets. Second, considering the availability of different transport vehicles and their associated capacities and logistic costs, the DMs would like to decide about the most appropriate type and the number of transport vehicles required at each supermarket to deliver the prepared bins to the stations.

There are three types of transport vehicles, as shown in [Figure 1](#), where their capacity in terms of bins, their shipment, and procurement costs are represented in [Table A1](#). Also, the company is interested in knowing about the effect of different supermarket capacities, which can be a multiple of 10 starting from 30 bins (i.e., 30, 40, 50, etc.), on the above-mentioned decision problems. It is worth mentioning that in this study as per discussion with the company's experts, the installation cost (IC) of each supermarket is calculated by multiplying the supermarket capacity by 20, wherein their associated expenses such as equipment and establishment costs are taken into consideration. The rest of the data associated with the real case is shown in [Table A2](#). In determining the above cost term parameters, a fixed planning horizon (e.g., month or year) for the production process has to be considered. Considering that the supermarket installation and vehicles selection are strategic problems, average production over a couple of years is usually considered for informed decisions so that the fluctuations over longer periods are taken into consideration. In this study, yearly production as the planning horizon is considered for calculating the reasonable values of installation cost, procurement cost, and expected demand. Thus, at this strategic decision-making phase, the company attempts to simultaneously address the SLP and TVSP while the PF's total cost is minimized.

According to the above explanation, the problem can be formulated by assuming a set of stations, $k = 1, \dots, M$ (M = number of stations), arranged along a single straight AL with known part demands in bins as sd_k , which have to be supplied by a set of candidate supermarkets $s = 1, \dots, S$ (S = possible number of supermarkets). There are different types of transport vehicles $e = 1, \dots, E$ (E = number of transport vehicles) which can be used by supermarkets to deliver the bins to a sequence of stations. Each transport vehicle has a different capacity as well as shipment and procurement costs. The coordinate of station k and supermarket s on the AL are defined by (x_k, y_k) and (X_s, Y_s) , respectively. The distance that each transport vehicle travels from supermarket to supply bins to a sequence of stations, which is represented by $dist_{skl}$, is calculated by the following components: (a) the distance from supermarket s to station k , (b) the distance between station k to station l , and (c) the distance from station l to supermarket s . Equation (1) presents how $dist_{skl}$ can be calculated.

$$dist_{skl} = |X_s - x_k| + |Y_s - y_k| + |x_k - x_l| + |y_k - y_l| + |X_s - x_l| + |Y_s - y_l| \quad (1)$$

The integrated SLP and TVSP considered in this study, named by SLTVSP, aims to determine the optimum number and location of supermarkets and the stations that each support, as well as the type and number of transport vehicle required by each supermarket, so that the total cost of PF in terms of part shipment, transport vehicle procurement, and supermarket installation costs are optimized simultaneously. An AL layout with three supermarkets and three transport vehicles namely AGV, tow train and trolley, to supply the bins of parts to the shooter racks of stations is illustrated in Figure 1.

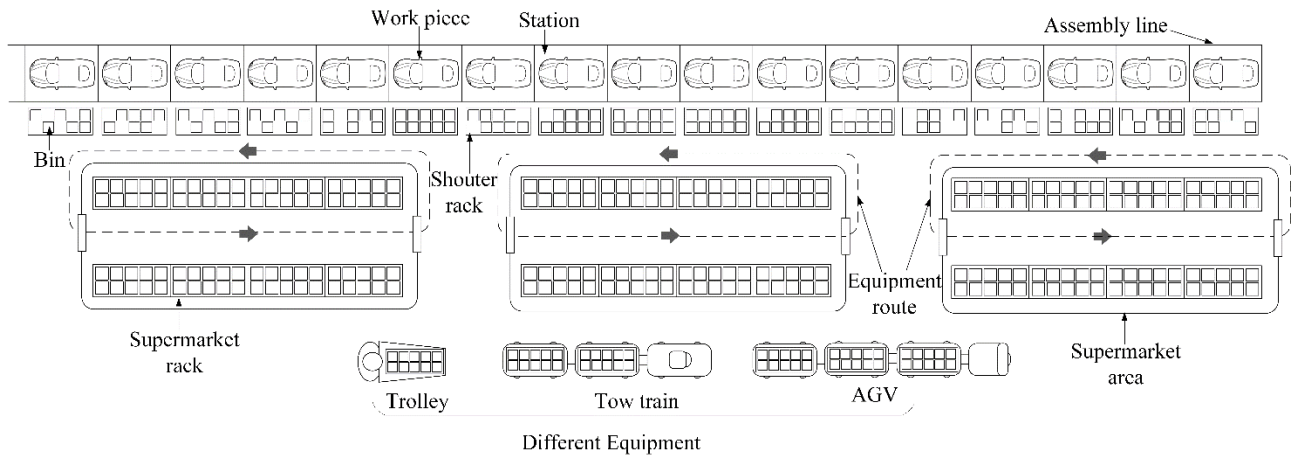


Figure 1. The layout of an AL with three supermarkets and three transport vehicles.

The following assumptions are considered in modeling the SLTVSP which are taken from the literature (Alnahhal and Noche, 2015; Emde and Boysen, 2012) as well as the observations made in the real case:

- There are candidate supermarket places located next to the shooter racks of stations arranged in a straight pattern in a single AL depending on the space limitation of the shop floor.
- The capacity of supermarkets is limited which is known in terms of the number of bins that can be stored and retrieved.
- There are different types of transport vehicles with different capacities in terms of bins. There are no limits on the number of vehicles of each type which can be used by each supermarket. Although there are different transport vehicles, however, it is assumed that each supermarket can only use one type of transport vehicles among the available types.
- The parts are re-packed and stored in bins of identical standardized sizes in the supermarkets to be delivered to the stations. Only frequent deliveries in small lots from supermarkets to assembly stations are considered in this study. Moreover, to enable the use of fully automated loading and unloading systems, the bins are considered to be identical in terms of size. It is worth mentioning that, the use of identical standardized size bins is a requirement for applying “shooter-racks” as observed in the case study and also discussed in previous research studies (e.g., Zhou and Tan, 2019).

- The transport vehicles set off from each supermarket and visit a sequence of stations, consecutively. For instance, if the sequence of stations from 1 to 4 have to be supplied by a supermarket, all the stations in this range will be visited sequentially by the same assigned transport vehicle to receive their parts. Thus, it is required to have all the stations (re)arranged in increasing order.

3.2. Proposed model

The notations used in modeling the considered SLTVSP are presented in [Table 2](#).

Table 2. List of notations.

Notation	Definition
Indices:	
k, l :	Index for stations ($k, l = 1, \dots, M$)
s :	Index for supermarket ($s = 1, \dots, S$)
e :	Index for transport vehicle ($e = 1, \dots, E$)
Parameters:	
M :	Number of stations
S :	Possible number of supermarkets
E :	Number of transport vehicles
x_k, y_k :	Coordinates x and y for station k
X_s, Y_s :	Coordinates x and y for supermarket s
$dist_{skl}$:	Total distance traveled by transport vehicle from supermarket s to supply all stations from stations k to l ;
sd_k :	Demand of station k in terms of number of bins
$Tdem_{kl}$:	Total demand of all stations from station k to station l ; $Tdem_{kl} = \sum_{f=k}^l sd_f$
Dsd_{kl} :	Total standard deviation of part demand for all stations from stations k to l
$Caps$:	Capacity of supermarket (number of bins)
IC :	Installation cost of supermarket
$Cape_e$:	Capacity of transport vehicle e (number of bins)
$Proce_e$:	Procurement cost for one transport vehicle of type e
SC_e :	Shipment unit cost of moving one bin one unit distance using transport vehicle e
α :	Service level for probability of not exceeding the capacity of the supermarket
Decision variables:	
NS :	Number of established supermarkets
q_{se} :	Number of transport vehicle e assigned to supermarket s
z_{skle} :	$\begin{cases} 1; & \text{if all stations from station } k \text{ to } l \text{ are fed by supermarket } s \text{ using transport vehicle } e \\ 0; & \text{otherwise} \end{cases}$

According to the problem description given above, a MIP model for SLTVSP is proposed as follows. It is worthy of mentioning that this model is an extended version of the model proposed by Alnahhal and Noche (2015) by assuming that different transport vehicles can be used by supermarkets. As a result, in the objective function, the shipment and vehicle procurement costs are added/updated. In the constraint, a new constraint is added to calculate the minimum number of transport vehicles. Moreover, the rest of the constraints and the decision variables are updated to be able to take the difference in transport vehicles into considerations. This model is generic in the sense that it can be applied to jointly address two

main decision problems of PF of ALs using supermarkets (SLP and TVSP), while different supermarket locations and transport vehicles are considered.

$$\text{Min } TC = \sum_{s=1}^S \sum_{k=1}^M \sum_{l=k}^M \sum_{e=1}^E z_{skle} \times Tdem_{kl} \times dist_{skl} \times SC_e + \sum_{s=1}^S \sum_{e=1}^E q_{se} \times Proce_e + NS \times IC \quad (2)$$

S.T.:

$$q_{se} \geq \sum_{k=1}^M \sum_{l=k}^M z_{skle} \times \frac{Tdem_{kl}}{Cape_e} \quad \forall s = 1, \dots, S; \forall e = 1, \dots, E \quad (3)$$

$$NS = \sum_{s=1}^S \sum_{k=1}^M \sum_{l=k}^M \sum_{e=1}^E z_{skle} \quad (4)$$

$$\sum_{s=1}^S \sum_{k=1}^b \sum_{e=1}^E z_{skbe} = \sum_{s=1}^S \sum_{l=b+1}^M \sum_{e=1}^E z_{s(b+1)le} \quad \forall b = 1, \dots, M - 1 \quad (5)$$

$$\sum_{k=1}^M \sum_{l=k}^M \sum_{e=1}^E z_{skle} \leq 1 \quad \forall s = 1, \dots, S \quad (6)$$

$$(Tdem_{kl} + F_{\alpha} Dsd_{kl}) \times z_{skle} \leq Caps \quad \forall s = 1, \dots, S; \forall k = 1, \dots, M; \forall l = k, \dots, M; \forall e = 1, \dots, E \quad (7)$$

$$NS \geq 1 \quad (8)$$

$$z_{skle} \in \{0, 1\} \quad \forall s = 1, \dots, S; \forall k = 1, \dots, M; \forall l = k, \dots, M; \forall e = 1, \dots, E \quad (9)$$

$$NS, q_{se} \in \text{Integer} \quad \forall s = 1, \dots, S; \forall e = 1, \dots, E$$

Using Equation (2) the objective function of the model, which is the total cost (TC) of PF for SLTVSP, is minimized. The objective function is comprised of three terms, including the total shipment cost (TSC), total vehicle procurement cost ($TVPC$) and total installation cost (TIC) of supermarkets, respectively. Constraint (3) obtains the number of transport vehicles of type e assigned to each supermarket. Using constraint (4), it is assured that the number of station groups (cells) formed to be supplied by transport vehicles, amounts to the number of established supermarkets. By constraint (5), it is assured that all cells are supplied by the established supermarkets through the assigned transport vehicle type. Using constraint (6), it is guaranteed that each station is supplied by only one supermarket as well as one type of transport vehicle. Assuming that the total demand of each supermarket can be subjected to changes occasionally caused by the variations of demands between its associated station groups, constraint (7) assures that the capacity of the supermarket is adequate not just for the total demand of stations groups ($Tdem_{kl}$) but also for the added standard deviation of stations demand (Dsd_{kl}) multiplied by F_{α} , which is the normal factor for the α quantile of the standard normal distribution. Constraint (8) ensures that at least one supermarket is established to supply the stations' demand. Constraint (9) delimits the domain of the decision variables which are both binary and integer.

4. The proposed hybrid genetic algorithm for SLTVSP

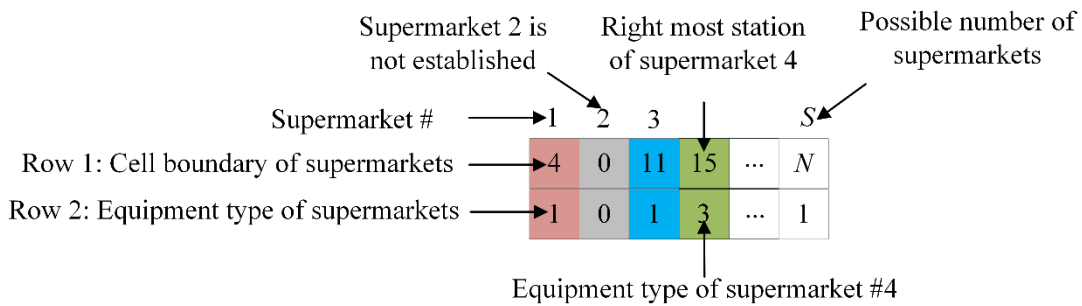
Due to the complexity of basic SLP (Alnahhal and Noche, 2015; Zhou and Tan, 2019), which even becomes more complex after integration with TVSP in this study, using approximation methods to solve large-sized problems is inevitable. In this regard, since GA has shown good

performance in addressing a large variety of optimization problems such as scheduling (Zhao et al., 2017), line balancing (Fathi et al., 2019) and SLP (Alnahhal and Noche, 2015), a new hybrid GA, called GA-VNS is proposed to deal with the SLTVSP. More details about the proposed GA and its elements are provided in the following sections.

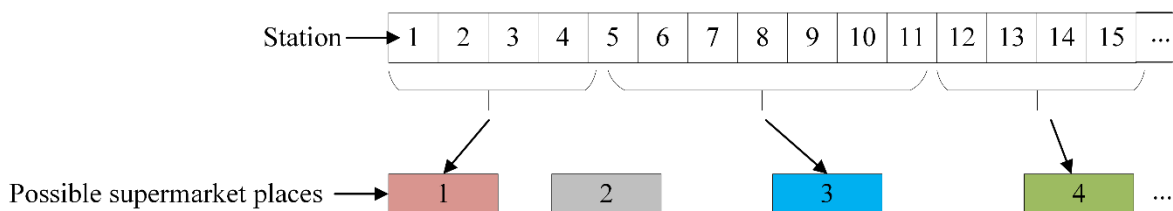
4.1. Representation and initialization

This study proposes a new solution representation for the proposed GA which is based on a two-row chromosome structure. The first row of the chromosome shows the supermarket positions ranged from one to the possible number of supermarkets contained by the cell boundary or the rightmost station assigned to each supermarket. For instance, in Figure 2 (a) the first position which belongs to supermarket 1, contains 4 as the rightmost station assigned to supermarket 1 meaning that stations 1-4 are assigned to supermarket 1. It is worth mentioning that using this representation when a supermarket is not established, e.g., supermarket 2 in Figure 2 (a), its cell boundary will be equal to zero.

On the other hand, the second row illustrates the transport vehicle type assigned to each established supermarket. For instance, supermarket 1 in Figure 2 (a) uses transport vehicle type 1 while the second supermarket is not established, and thus no transport vehicle is assigned. To better elaborate this representation, Figure 2 (b) illustrates the corresponding solution for the representation shown in Figure 2 (a) which depicts the assignment of stations to the possible supermarket places. As can be seen from this figure, stations 1-4 are assigned to supermarket 1 which are colored similar to Figure 2 (a), supermarket 2 is not established, etc.



(a) An example of two-row chromosome representation



(b) The solution corresponding to the above representation

Figure 2. The representation scheme for GA.

By assuming the population size of GA as $Popsiz$, the initial population of GA is composed of a matrix with the dimension of $(2 \times Popsiz, S)$. To make sure that the search space of the SLTVSP is efficiently searched by the GA-VNS, the following initial solution generation procedure is proposed to obtain a feasible initial solution for SLTVSP:

Procedure feasible SLTVSP solution generation

```

Begin
 $s = 1; k = 1;$ 
While  $s \leq S$  and  $k \leq K$ 
  If  $\sum_{k \in s} sd_k + F_\alpha \times Std_{k \in s}(sd_k) \leq Caps;$ 
     $W_s \leftarrow k$ 
  Else
     $s = s + 1$ 
  End if;
End while;
End procedure.

```

where Std. is the standard deviation and W_s is the set of stations assigned to supermarket s and the rest of notations are the same as presented in [Table 2](#).

4.2. Crossover

According to the proposed representation, a new crossover operator is also devised. The crossover operator selects two individuals (parents) from the current population randomly and then uses the crossover operation illustrated in [Figure 3](#) to make two new offspring. As this figure shows, the parents have 11 genes which equal to the possible number of supermarkets in this example. Then a random crossover point is selected for both of the selected parents. As shown in [Figure 3](#), three combinations of crossover points (A, B, and C) are considered for further elaboration. Unlike the usual crossover operators in which the lengths of the selected parents are equal, in this crossover operator the lengths of the selected parents can be different from each other. For example, in [Figure 3](#) parent 1 and 2 have different lengths of 10 and 11, respectively, which is the place where the last station (i.e., 21 in this example) appears. It is important to note that while doing the crossover, the zero values are ignored and only positive numbers are taken into considerations. In order to make sure that during the crossover operations, the characteristics of the parents are shared between the two offspring while the feasibility of the new offspring is maintained, both of the following two conditions have to be satisfied.

First, the exchange of the genes after the crossover points should not violate the sequence of the group of cells which has to be in ascending order. For instance, in crossover combination A labeled with A in [Figure 3](#), since the maximum number to the left (inclusive) of point A in parent 1 (10) is lower than the minimum number to the right of point A in parent 2 (12), shown by oval $10 < 12$, the ascending order condition for offspring 1 is met. By making the same comparison for offspring 2 since $6 < 15$, the ascending order condition for offspring 2 is also met. Thus, the crossover variant A is valid when considering the first condition. However, since for the crossover combination B, this condition is not met for offspring 1 ($4 \not< 3$), thus this crossover is not valid.

The second condition for a successful crossover is that while the first condition is satisfied, the length of the resulting offspring should not be longer than the possible number of supermarkets. For instance, in crossover variant C in the last row of Figure 3, the first condition is met for both of the resulting offspring, while the length of offspring 2 is larger than the possible number of supermarkets which is 11 in this example. Thus, this crossover combination is not valid, too.

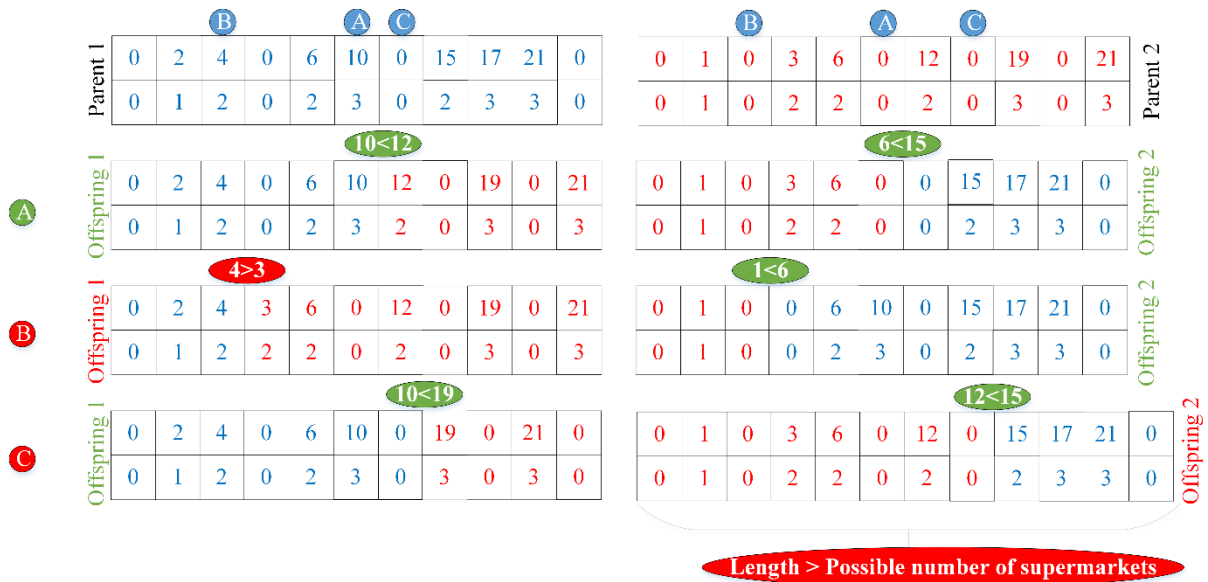


Figure 3. Crossover operator for GA.

It is worthy of mentioning that using the above crossover, while exchanging the elements of parents in the first rows, the transport vehicle type of the supermarkets are also altered in the second row at the same time as depicted in Figure 3. The above crossover operation is performed on the individuals of the current population, considering the crossover probability (Pr_{cr}).

4.3. Mutation

Regarding the mutation operator adopted for the proposed GA, the aim is to change the number within a selected mutation point so that its left and right cell boundaries are not exceeded. To do so, an individual from the current population is randomly selected and a random mutation point is chosen. For this mutation point, a set of candidate numbers considering its left and right cell boundaries are built. If the mutation point contains a positive number, then a zero is also added to the above set of candidate numbers (a zero means that the supermarket is not established) and then a number from the built candidate set is randomly chosen. For instance, in Figure 4, for the mutation point A, the candidate set is {0, 2, 3, 4, 6} when considering its left and right cell boundaries. For mutation point B in Figure 4, which includes zero, the candidate set from which a number has to be chosen randomly is {11, 12, 13, 14}. It is important to emphasize that whenever a mutation operation is performed on the

first row, a number from $[1, \dots, E]$ has to be chosen randomly for the second row of the corresponding mutation point. Since in this example, three transport vehicle types were considered, this range is $\{1, 2, 3\}$ as shown in Figure 4.

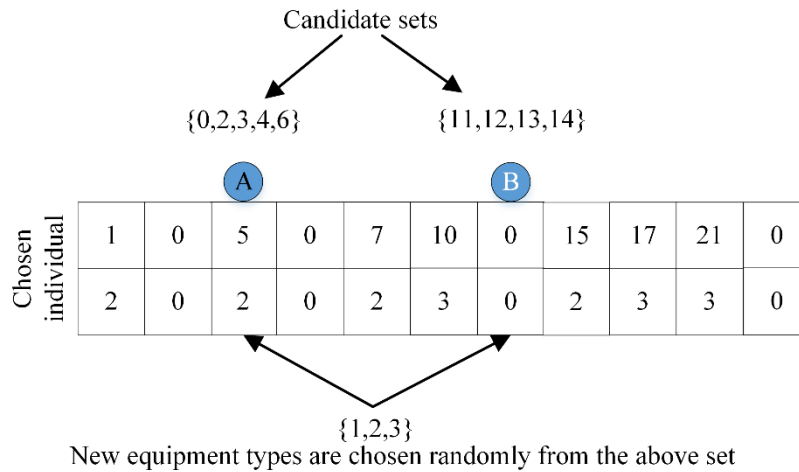


Figure 4. Mutation operator for GA.

The above mutation operation is performed on the current population's individuals, considering the mutation probability (Pr_{mu}).

4.4. Variable neighborhood search

The reason for applying the variable neighborhood search (VNS) is to systematically use different search structures to improve the possibility of finding a better solution (Mladenovic and Hansen, 1997; Zhao et al., 2017). Using the crossover and the mutation operators, the diversity of the search in terms of the assignment of different stations to different supermarkets is assured. However, to make sure that there is a specific mechanism in the algorithm to guide the search in terms of transport vehicles selection, a new operation is required. In this study, VNS is employed to enhance the local search capability of the GA which is mainly responsible for guiding the GA with transport vehicles selection. To this purpose, two neighborhood operators (2-opt and 3-opt), adapted from the literature (Liu et al., 2018; Zhao et al., 2017), are used in which the two and three points random swaps of genes are performed on the second row of each individual (chromosome) which can, in turn, increase the diversity of the solutions while improving the search capability of the proposed GA by avoiding being trapped in the local optimum. The pseudo-code of the VNS applied in each iteration of the proposed GA is shown in the following procedure:

Procedure VNS

Select a set of the best individuals from the current population

For each individual (x)

Repeat

Obtain local optimum for x using 2-opt neighborhood operator (x')

If this local optimum is better than the current solution: $x \leftarrow x'$

Else obtain local optimum for x using 3-opt neighborhood operator (x'')

If this local optimum is better than the current solution: $x \leftarrow x''$

End repeat

End procedure.

The above VNS operation is performed on the best individuals of the current population considering the VNS probability (Pr_{VNS}).

Overall, the mechanisms discussed above result in a new GA-VNS for SLTVSP. The pseudo-code of the proposed algorithm for SLTVSP is summarized as follows:

Pseudo-code of GA-VNS for SLTVSP

Initialization step

Read input parameters: $sd_k, S, Caps, IC, Proce_e, SC_e, F_\alpha, x_k, y_k, X_S, Y_S$;
Calculate: $Tdem_{kl}, Dsd_{kl}, dist_{skl}$;

Pre-processing step

Set values for the GA control parameters ($Popsi\text{ze}, Pr_{cr}, Pr_{mu}, Pr_{vns}, Max\ Gen$);
Initialize the generation counter ($G = 0$);
Use the representation and initialization mechanisms in 4.1 to generate a population of initial feasible solutions;
Calculate the TC of the current population's solutions

Repeat

For $i = 1$ to $Popsi\text{ze}$ do (create population P of the next generation)

Begin

Reproduction step

Rank the solutions within the current population according to their TC s and considering the reproduction rate put the best found solutions as the next population members;

Crossover step

Considering the Pr_{cr} for each of the two randomly selected individuals from the current population apply the crossover scheme in 4.2 to generate the next population individuals;

Mutation step

Considering the Pr_{mu} for each randomly selected individual from the current population apply the mutation scheme in 4.3 to mutate the next population individuals;

Evaluation step

Calculate the TC for each SLTVSP solution using Equation (2);

End;

Variable neighborhood search step

Rank the solutions within the current population according to their TC s and perform the VNS operation in 4.4 on the best solutions of the current population considering the Pr_{vns} ;

Update the best found solution

Update the best found solution

$G = G + 1$;

Until the stopping condition has been reached;

Return the best found solution for the SLTVSP.

5. Computational study

In this section, the computational settings, SLTVSP and SLP results, and their comparison are reported and discussed in detail.

5.1. Computational settings

The MIP model for SLTVSP was coded in GAMS version 24.1.2 and solved with the CPLEX solver. To present the performance of the proposed GA-VNS, it is compared with conventional GA as well as SA. It should be noted that due to the complexity of MIP, the computational time of exact algorithms increases exponentially. Thus, the computational time for CPLEX was limited to 1,200 seconds. The GA-VNS, GA, and SA were coded in MATLAB

version R2015a. Both the MIP model and the meta-heuristics algorithms were run on a personal computer with a Core i7 2.4 GHz processor and 8 GB of RAM.

To evaluate the performance of the CPLEX solver, GA-VNS, GA and SA for the proposed MIP model, since the model is new, and no benchmark problems exist in the literature, in addition to the real case study introduced, a set of 19 benchmark problems are generated and solved using the CPLEX solver and the above-mentioned meta-heuristic algorithms. It is important to pay notice that the test problems are generated with the same characteristics of the real-world case taken from industry and can be downloaded at the link provided in the [Data reference Section](#). All the considered problems are divided into three problem sizes based on the number of stations, namely, small (20 to 60 stations), medium (70 to 130 stations) and large (140 to 200 stations) as discussed with the experts in the field. The remaining characteristics of the generated problems are discussed as follows.

In line with the problem description, it is presumed that the stations are sequentially arranged along a straight AL with one unit distance from each other. For simplicity, the x -coordinates of candidate supermarket places (X_s) are chosen to be smoothly positioned in the range of the locations of the stations. Moreover, the y -coordinates of stations and supermarkets (y_k and Y_s) are set to 0 and 5, respectively (Nourmohammadi et al. 2019). Based on the requirement in the real case, the information about the transport vehicle types as well as the supermarkets was chosen according to the problem description section.

To set reasonable parameter values for GA-VNS, Taguchi design was applied on an $L_9(3^4)$ experimental design (Taguchi et al. 2007) over its parameters' levels, as shown in [Table 3](#).

Table 3. GA-VNS parameters' levels.

Parameter level	$Popsi\text{ze}$	Pr_{cr}	Pr_{mu}	Pr_{VNS}
1	20	0.7	0.1	0.05
2	50	0.8	0.15	0.1
3	100	0.9	0.2	0.2

According to the Taguchi results which can be accessed in the [supplementary material](#), the GA-VNS parameters levels are set to 100, 0.8, 0.2, and 0.2 for $Popsi\text{ze}$, Pr_{cr} , Pr_{mu} , and Pr_{vns} , respectively.

Since this study aims to compare the performance of the proposed GA-VNS with other meta-heuristics and due to the lack of meta-heuristics in this context, particularly with the characteristics of the new SLTVSP discussed in this study, a conventional GA with the same mutation and crossover operations will be considered further for comparison purpose. To determine the effect of the VNS mechanism in the proposed GA-VNS and for a fair comparison, the levels of the GA parameters are also set equal to the levels of the GA-VNS tuned by Taguchi. In addition, simulated annealing (SA) is applied to SLTVSP by embedding the proposed mutation operator in Section 4.4 as the neighborhood search mechanism.

For a fair comparison in terms of an equal number of function evaluations (NFE) for all algorithms, the following stopping conditions are considered. The stopping condition of GA-

VNS was chosen to be 500 generations. Using the population size of 100 based on Taguchi design, the resulting NFE for GA-VNS by taking the 2 VNS operators and the selected Pr_{vns} level (20%) into consideration, is at most 70,000. Thus, for a fair comparison in terms of equal NFE, the stopping condition of GA was set to 700 generations (by assuming the same $Popsiz$ as GA-VNS). Accordingly, the initial temperature of SA was set to 70,000, and the stopping condition of SA was set to reaching the final temperature of zero while using the cooling scheme $T_{IT+1} = T_{IT} - 1$, where T_{IT} is the temperature at IT th iteration. Also, the internal iteration number of SA at each temperature was set to maximum 5 iteration, while the acceptance rate of the SA at each iteration is calculated using Equation (10) (Fathi et al. 2016).

$$Acceptance\ rate_{IT} = \exp\left(\frac{\Delta TC}{T_{IT}}\right) \quad (10)$$

where ΔTC is the change in the objective function value between two solutions.

5.2. Computational results

In this section, the results of the mathematical model and the meta-heuristic approaches over the SLTVSP test problems are compared against one another. To consider the stochastic nature of the meta-heuristic algorithms, they are run over each problem 10 times and their results are reported in terms of TC of PF including shipment, transport vehicle procurement and installation costs of supermarkets. Table 4 shows the results of CPLEX, GA-VNS, GA, and SA over the case study (the first two rows with 45 stations) as well as different test problems. The first four columns of Table 4 show the characteristics of the problems, including the problem size, name, number of stations (M) and the capacity of the supermarkets ($Caps$). The name selected for each problem represents the combination of the number of stations (M) and the capacity of supermarkets. Column S shows the possible number of supermarkets which is calculated by $(Tdem_{1K} + F_{\alpha}Dsd_{1K})/Caps$ added by 3 to obtain a reasonable upper bound for each problem (Alnahhal and Noche, 2015). Under column CPLEX, the optimal or best-found solution, the optimality gap (OG) in percent and the time spent (in seconds) to find the optimal or the best TC within the allotted time (1200 second), are reported. In the rest of the columns, the worst, the average, the best and the standard deviation of the resulting TCs for each meta-heuristic, as well as the average time (in seconds) spent by them to find the optimal or best-found solution are represented.

According to Table 4, one can observe that in total, the worst, average, and best TCs obtained by GA-VNS outperform the relating TCs obtained by GA and SA. Also, the standard deviations of the TCs obtained by GA-VNS are mostly less than GA and SA, which can be a representative measure indicating the robustness and reliability of GA-VNS compared to the other methods (Osaba et al., 2016), particularly when applying it in the real environment. For the case study (the first two rows), in both of the considered capacities, the optimal solutions have been found by both MIP and GA-VNS, while GA and SA could only find near-optimal and feasible solutions, respectively. Detailed solutions obtained by GA-VNS for all the solved problems are given in the Data Reference.

Furthermore, to better clarify the deviations between the TC s obtained by each algorithm, i.e., GA-VNS, GA and SA, and the optimal or best found TC s obtained by CPLEX, a new measure called relative percent deviation (RPD) is calculated using Equation (11). Table 5 shows the RPD_{worst} , RPD_{Avg} , and RPD_{best} for each algorithm over different test problems.

$$RPD(\%) = \frac{TC_{Algorithm} - TC_{CPLEX}}{\min(TC_{CPLEX}, TC_{Algorithm})} \times 100 \quad (11)$$

According to Table 5, one can observe that in 9 out of 40 problems, both GA-VNS and MIP have resulted to the same optimal or best-found solutions, while in 21 problems, the best TC s obtained by GA-VNS have less than one percent deviation from the optimal or best solutions found by CPLEX. For 7 problems, the GA-VNS solutions have shown between one to two percent deviations from the optimal or best solutions found by CPLEX. Finally, it is important to highlight that in three problems belonging to the large-sized problems the RPD_{best} obtained by GA-VNS were negative, meaning that the best TC s resulted by GA-VNS were better than their relating TC s obtained by CPLEX, during the allotted time. This superiority was made possible because of the positive optimality gap (OG%) of the CPLEX in the large-sized problems.

In Figure 5, the comparison of GA-VNS with MIP, in terms of the CPU time and the average RPD_{best} for three problem sizes (small, medium, and large), is shown. It can be seen that when the size of the problem increases, the difference between the CPU times of MIP and GA-VNS has increased dramatically due to the complexity of the problem. Moreover, it is interesting to note that the average RPD_{best} for large-sized problems (0.34) is lower than the small (0.51) and the medium (0.54) sized problems.

In addition, Figure 6 illustrates the comparison of MIP CPU times with the average CPU times for different meta-heuristics to reach optimal or best-found solutions. As this figure shows, when the size of test problems increases, the computational time of MIP has grown exponentially so that it is unable to find the optimal solution in 12 out of 14 problems belonging to large-sized problems. This is because meta-heuristics are less sensitive to the size of the problems. Moreover, as this figure shows the amount of CPU times spent by MIP over most of the test problems, which are shown in logarithmic scale with base 10, are much higher than GA-VNS, GA, and SA, particularly for large-scale problems. Thus, by considering the performance of GA-VNS in achieving very near-optimal solutions, i.e., smaller RPD_{best} with shorter average CPU times (as shown in Figures 6), one can infer that it is advantageous to apply GA-VNS in addressing the SLTVSP.

Table 4. Computational results of CPLEX, GA-VNS, GA and SA in terms of TC .

Prob. Size	Problem Name	M	Cap_s	S	CPLEX			GA-VNS				GA				SA						
					OP/BFS*	OG(%)	Time (Sec.)	Worst	Avg.	Best	S. Dev.	Time (Sec.)	Worst	Avg.	Best	S. Dev.	Time (Sec.)	Worst	Avg.	Best	S. Dev.	Time (Sec.)
Small	M45-Cap30	45	30	14	22001	0.00	4**	22297	22223	9734	109	6	22533	22240	22023	165	6	25419	24842	24197	383	16
	M45-Cap50		50	10	23324	0.00	45	23726	23686	11232	121	4	23726	23726	23726	0	5	30736	28225	26893	1245	12
	M20-Cap30	20	30	8	9734	0.00	2	9751	9749	16402	5	2	9751	9748	9734	6	1	10872	10377	10094	350	5
	M20-Cap50		50	6	11232	0.00	2	11496	11353	18298	99	4	11496	11291	11232	96	2	13201	12236	11531	692	0
	M30-Cap30	30	30	12	16402	0.00	1	16697	16637	18730	84	2	16897	16719	16632	83	2	19208	18370	17444	441	17
	M30-Cap50		50	8	18098	0.00	2	18298	18298	19868	0	2	18298	18298	18298	0	1	21310	19703	19368	606	10
	M40-Cap30	40	30	12	18594	0.00	3	18730	18730	22001	0	3	18748	18739	18730	9	2	21408	20635	20175	383	14
	M40-Cap50		50	9	19868	0.00	11	20216	20088	23324	112	11	20784	20317	20120	249	4	24565	23298	21780	1093	9
	M50-Cap30	50	30	16	26243	0.00	4	26455	26373	26339	45	9	26617	26403	26327	100	3	29011	28526	28036	322	21
	M50-Cap50		50	11	28168	0.00	14	29048	28896	28678	128	7	29048	28853	28660	181	3	33289	31813	30439	1061	12
M60-Cap30	60	30	17	26680	0.00	12	26998	26938	26796	52	7	27588	27035	26862	258	4	32863	31693	30558	730	15	
M60-Cap50		50	11	28676	0.00	88	29782	29338	29168	263	7	29644	29350	29028	242	3	35583	33690	32404	1145	15	
Medium	M70-Cap30	70	30	20	33390	0.00	13	33863	33646	33547	143	15	34258	33966	33699	196	8	38154	37465	36635	480	16
	M70-Cap50		50	13	35620	0.00	34	35973	35847	35620	149	3	37036	36679	36348	276	5	45699	43974	41221	1408	12
	M80-Cap30	80	30	22	35869	0.00	17	36013	35919	35869	63	19	36925	36385	35869	374	7	48522	46783	44451	1222	23
	M80-Cap50		50	14	39518	0.00	111	40726	40151	39662	467	11	41163	40958	40726	172	7	51733	49043	46921	1699	14
	M90-Cap30	90	30	24	41511	0.00	143	42355	42103	41891	156	29	43633	42915	41903	515	16	49303	47969	46952	749	25
	M90-Cap50		50	16	44813	0.00	806	46196	46089	45588	200	14	47648	46500	46176	541	8	58799	56455	54867	1298	19
	M100-Cap30	100	30	26	44279	0.00	168	44689	44417	44351	98	24	45733	45228	44880	332	19	53493	52118	51067	821	25
	M100-Cap50		50	17	48351	0.00	192	48913	48696	48959	188	19	50827	50524	50118	263	9	68165	60945	55409	3171	16
	M110-Cap50	110	50	19	57122	2.33	1200***	58236	57566	57446	273	8	59310	58653	58246	391	9	79641	75471	72046	2285	22
	M110-Cap60		60	16	54612	0.00	57	54962	54781	54612	106	20	55484	54893	54730	274	8	78914	76356	71784	2012	19
	M120-Cap50	120	50	20	58339	0.00	336	61054	59180	58641	872	7	62742	61756	61054	596	7	80284	76572	72255	2315	20
	M120-Cap60		60	17	57265	0.00	214	57696	57666	57639	24	14	60597	58864	57674	883	7	88192	79891	69662	5908	18
M130-Cap50	130	50	22	67435	0.00	461	70445	68689	67583	1159	15	71559	71016	70544	364	13	111374	99468	93374	5076	25	
M130-Cap60		60	19	66059	0.00	532	67363	67104	66528	198	21	69228	68189	66998	763	10	104385	95766	90853	3906	21	
Large	M140-Cap50	140	50	22	69168	0.31	1200	71748	69954	69924	901	12	73120	72423	71748	439	10	109905	104160	96579	4054	23
	M140-Cap60		60	19	67359	0.00	348	68402	67848	67736	189	18	69438	68190	67736	652	10	113875	102436	91900	6289	20
	M150-Cap50	150	50	24	74229	1.28	1200	76200	74822	74785	535	41	79955	78013	76885	934	15	116504	108416	103683	4260	27
	M150-Cap60		60	20	72120	0.00	601	73573	73197	73121	135	23	76093	74331	73121	1080	11	123073	114011	105709	4880	25
	M160-Cap50	160	50	25	78198	0.83	1200	79064	78699	78506	217	23	84685	82952	81904	1023	19	127470	122896	118604	2418	20
	M160-Cap60		60	21	76700	0.90	1200	77555	77477	76810	207	20	79466	78673	77823	554	10	133970	122810	115188	5743	24
	M170-Cap50	170	50	27	84509	0.63	1200	85377	85126	84669	265	6	92149	90138	89111	1046	17	122175	117518	111784	2915	25
	M170-Cap60		60	23	82808	1.26	1200	83540	83156	82844	290	26	87674	85300	83568	1481	15	142432	134396	122153	5603	19
	M180-Cap50	180	50	28	88477	3.61	1200	89294	88904	88357	331	7	94200	93277	92262	768	19	151377	141497	129364	6567	25
	M180-Cap60		60	24	86960	2.17	1200	88075	87705	87207	199	25	93241	90858	89453	1352	13	159110	142587	128591	9033	26
	M190-Cap50	190	50	28	89136	5.04	1200	90016	89307	89134	258	4	94955	94380	93899	380	19	154755	146245	137632	5268	32
	M190-Cap60		60	24	86478	1.84	1200	87138	86874	86478	323	12	91352	89999	88914	901	16	158170	144946	127983	7478	23
	M200-Cap50	200	50	30	97932	3.05	1200	98304	98000	97924	152	15	107006	104921	103961	1023	20	148464	142456	134229	4663	26
	M200-Cap60		60	26	95919	1.72	1200	96419	96119	95919	245	8	101373	100539	98107	1122	14	163212	153245	138186	6694	28

*Optimal/Best found Solution; **Optimal solution has been found before reaching 1200 seconds; *** The best solution found during 1200 seconds (if CPU<1200 the solution is optimal, if CPU=1200 the solution is the best found solution).

Table 5. Comparison of RPDs for GA-VNS, GA and SA (in percent).

Prob. Size	Prob. Name	M	$Caps$	S	GA-VNS			GA			SA		
					Worst	Avg.	Best	Worst	Avg.	Best	Worst	Avg.	Best
Small	M45-Cap30	45	30	14	0.17	0.16	0,00	0.17	0.15	0.00	11.7	6.61	3.69
	M45-Cap50		50	10	2.35	1.08	0,00	2.35	0.53	0.00	17.53	8.94	2.66
	M20-Cap30	20	30	8	1.80	1.43	0,00	3.02	1.94	1.41	17.11	12.00	6.35
	M20-Cap50		50	6	1.11	1.11	1,11	1.11	1.11	1.11	17.75	8.87	7.02
	M30-Cap30	30	30	12	0.73	0.73	0,73	0.83	0.78	0.73	15.13	10.98	8.50
	M30-Cap50		50	8	1.75	1.11	0,00	4.61	2.26	1.27	23.64	17.26	9.63
	M40-Cap30	40	30	12	1.34	1.01	0,00	2.42	1.09	0.10	15.54	12.91	9.98
	M40-Cap50		50	9	1.72	1.55	0,00	1.72	1.72	1.72	31.78	21.01	15.30
	M50-Cap30	50	30	16	0.81	0.50	0,37	1.42	0.61	0.32	10.55	8.70	6.83
	M50-Cap50		50	11	3.12	2.58	1,81	3.12	2.43	1.75	18.18	12.94	8.06
Medium	M60-Cap30	60	30	17	1.19	0.97	0,43	3.40	1.33	0.68	23.18	18.79	14.54
	M60-Cap50		50	11	3.86	2.31	1,72	3.38	2.35	1.23	24.09	17.49	13.00
	M70-Cap30	70	30	20	1.42	0.77	0,47	2.60	1.72	0.93	14.27	12.20	9.72
	M70-Cap50		50	13	0.99	0.64	0,00	3.98	2.97	2.04	28.29	23.45	15.72
	M80-Cap30	80	30	22	0.40	0.14	0,00	2.94	1.44	0.00	35.28	30.43	23.93
	M80-Cap50		50	14	3.06	1.60	0,36	4.16	3.64	3.06	30.91	24.10	18.73
	M90-Cap30	90	30	24	2.03	1.43	0,92	5.11	3.38	0.95	18.77	15.56	13.11
	M90-Cap50		50	16	3.09	2.85	1,73	6.33	3.77	3.04	31.21	25.98	22.44
	M100-Cap30	100	30	26	0.92	0.31	0,16	3.28	2.14	1.36	20.81	17.70	15.33
	M100-Cap50		50	17	1.16	0.71	1,26	5.12	4.50	3.65	40.98	26.05	14.60
Large	M110-Cap50	110	50	19	1.95	0.78	0,57	3.83	2.68	1.97	39.42	32.12	26.13
	M110-Cap60		60	16	0.64	0.31	0,00	1.60	0.51	0.22	44.50	39.82	31.44
	M120-Cap50	120	50	20	4.65	1.44	0,52	7.55	5.86	4.65	37.62	31.25	23.85
	M120-Cap60		60	17	0.75	0.70	0,65	5.82	2.79	0.71	54.01	39.51	21.65
	M130-Cap50	130	50	22	4.46	1.86	0,22	6.12	5.31	4.61	65.16	47.50	38.47
	M130-Cap60		60	19	1.98	1.58	0,71	4.80	3.22	1.42	58.02	44.97	37.53
	M140-Cap50	140	50	22	3.73	1.14	1,09	5.71	4.71	3.73	58.90	50.59	39.63
	M140-Cap60		60	19	1.55	0.73	0,56	3.09	1.23	0.56	69.06	52.07	36.43
	M150-Cap50	150	50	24	2.66	0.80	0,75	7.71	5.10	3.58	56.95	46.06	39.68
	M150-Cap60		60	20	2.01	1.49	1,39	5.51	3.07	1.39	70.65	58.09	46.57
Large	M160-Cap50	160	50	25	1.11	0.64	0,39	8.30	6.08	4.74	63.01	57.16	51.67
	M160-Cap60		60	21	1.11	1.01	0,14	3.61	2.57	1.46	74.67	60.12	50.18
	M170-Cap50	170	50	27	1.03	0.73	0,19	9.04	6.66	5.45	44.57	39.06	32.28
	M170-Cap60		60	23	0.88	0.42	0,04	5.88	3.01	0.92	72.00	62.30	47.51
	M180-Cap50	180	50	28	0.92	0.48	-0,14	6.47	5.43	4.28	71.09	59.93	46.21
	M180-Cap60		60	24	1.28	0.86	0,28	7.22	4.48	2.87	82.97	63.97	47.87
	M190-Cap50	190	50	28	0.99	0.19	-0,002	6.53	5.88	5.34	73.62	64.07	54.41
	M190-Cap60		60	24	0.76	0.46	0,00	5.64	4.07	2.82	82.90	67.61	48.00
	M200-Cap50	200	50	30	0.38	0.07	-0,01	9.27	7.14	6.16	51.60	45.46	37.06
	M200-Cap60		60	26	0.52	0.21	0,00	5.69	4.82	2.28	70.16	59.76	44.06

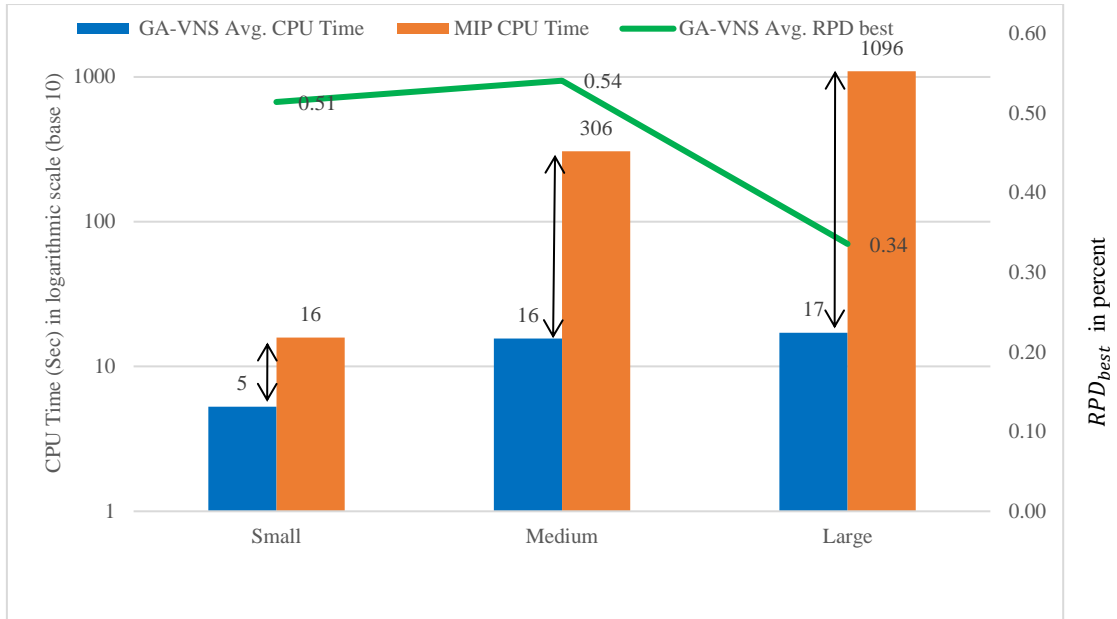


Figure 5. Comparison of MIP and GA-VNS in terms of CPU times and average RPD_{best} for three problems sizes.

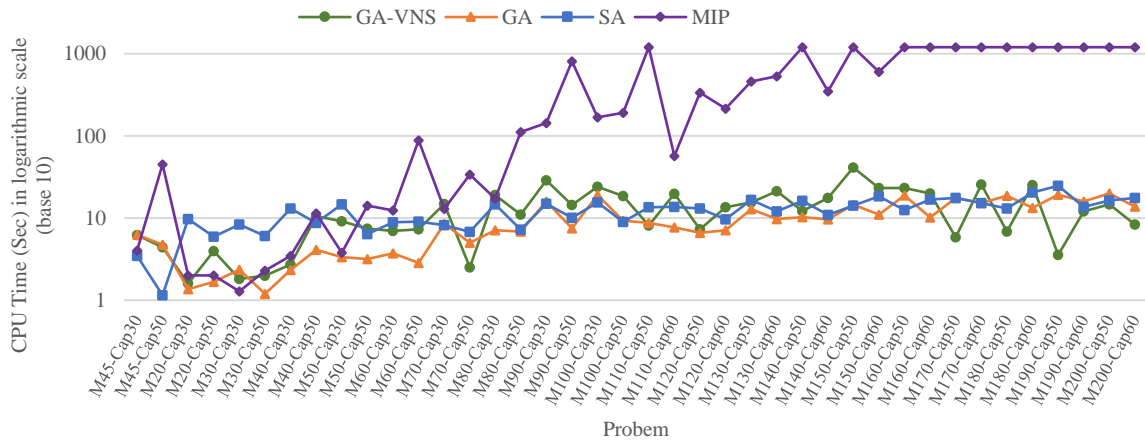


Figure 6. Comparison of MIP CPU times with average CPU times for GA-VNS, GA, and SA to reach the optimal or best found solutions.

To further validate the performance of GA-VNS, a statistical analysis of the results by the considered meta-heuristic algorithms is performed which can be accessed in the [supplementary material](#). According to the results of this statistical analysis, it can be concluded that the robustness and efficiency of GA-VNS in achieving near-optimal solutions are superior to other algorithms.

5.3. Comparison of SLTVSP and SLP results

To investigate the effect of TVSP on SLP, here the results of the MIP obtained for SLTVSP and SLP are compared. In this regard, the effect of implementing three different transport

vehicle types versus identical transport vehicles, i.e. when only tow train is available as the transport vehicle, while addressing the SLP is evaluated. Tow train is considered for SLP mainly because it is the most commonly applied transport vehicle in the literature (Emde and Gendreau 2017; Battini et al. 2013; Boysen et al. 2015).

Table 6 shows the average relative percent improvement (*RPI*) obtained by SLTVSP compared to SLP for the case study and the three problem sizes as well as in overall (including all problems), in terms of total installation cost (*TIC*), total vehicles procurement cost (*TVPC*), total shipment cost (*TSC*) and total cost (*TC*)¹. The positive values in Table 6 show an improvement in the related costs, while the negative values show a deterioration. For instance, in the case study, there are 7% improvement in *TIC*, 27% deterioration in *TVPC*, 37% improvement in *TSC*. Overall, for the case study, using different transport vehicles in SLTVSP has resulted in 8% improvement in *TC* compared to identical transport vehicles in SLP.

Table 6. Avg. *RPI* for SLTVSP compared to SLP in terms of different costs (in percent).

Problem	<i>TIC</i>	<i>TVPC</i>	<i>TSC</i>	<i>TC</i>
Case study	7	-27	37	8
Small	4	-25	39	9
Medium	4	-34	43	10
Large	6	-40	44	11
Total	4	-32	42	10

By moving from small to large problems, the amount of improvement in *TIC* has shown growth meaning that using SLTVSP has resulted in a lower number of supermarkets compared to SLP. On the other hand, due to an increase in the supermarket demand caused by a fewer number of supermarkets, there is a growing degrading trend in *TVPC* caused by implementing transport vehicles with higher capacities. In other words, from small to large-sized problems there is a trend to use transport vehicles with higher capacities and accordingly, higher procurement cost. However, in line with this trend, while moving from small to large-sized problems, the amount of saving and improvement in *TSC* has shown a growing trend caused by lower shipment costs for transport vehicles with higher capacities. Finally, by taking into account that *TC* is the main measure for managers in which a trade-off among the above different costs occurs, there is a growing improvement trend in *TC* when moving from small to large-scale problems. Overall, by comparing SLTVSP with SLP in total, there are 4% improvement in *TIC*, 32% deterioration in *TVPC*, 42% improvement in *TSC*, and finally 10% improvement in *TC*. Thus, it can be inferred that using different transport vehicles is beneficial for those managers who seek to optimize the *TC* of PF, which can be achieved through addressing the integrated SLTVSP.

6. Conclusions

Recently, many manufacturers have adopted the so-called supermarket to allow flexible and reliable just-in-time parts feeding (PF) to their ALs. However, to be able to apply supermarkets into practice, two primary and interrelated decision problems namely supermarket location

¹ $TC = TIC + TVPC + TSC$.

problem (SLP) and transport vehicles selection problem (TVSP) have to be addressed, wherein SLP aims to determine the number and locations of supermarkets while TVSP determines the type and the number of transport vehicles assigned to each supermarket. In this study, a mixed-integer programming (MIP) model for the integrated SLP and TVSP (called as SLTVSP), which was inspired from a real-world PF decision problem of an automotive manufacturing company, was proposed so that the total cost (*TC*) of PF, including the part shipment, transport vehicles procurement and supermarkets installations costs are simultaneously optimized. Furthermore, due to the complexity of the problem, a genetic algorithm (GA) with custom representations, initialization, crossover and mutation mechanisms was proposed to deal with the large-sized problems. To improve the search capability of GA, it was hybridized with a variable neighborhood search (GA-VNS). The computational results on the real-world case and over several generated test problems verified that the proposed GA-VNS provides a good approximation of the exact solutions obtained by MIP at a much shorter computational time, particularly for the large-sized problems. To further validate the efficiency of GA-VNS, its performance was also compared with conventional GA and SA, and the analyses of results proved the superiority of GA-VNS over other considered meta-heuristics. Finally, the comparison of SLTVSP and SLP results showed that using different transport vehicles is beneficial for optimizing the *TC* of PF by addressing the integrated SLTVSP rather than SLP with identical transport vehicles.

In this study, it is assumed that there is no limitation on the number of different transport vehicles that can be applied at each supermarket. However, in practice, there might be restrictions on the number of transport vehicles available at each supermarket due to the labor, space, and transportation system considerations. Moreover, considering the inter-dependency of different decisions made regarding the corporate objectives, the impact of SLTVSP on other operational decision problems such as scheduling and loading problems can be further investigated. Also, unlike this study which assumed that the station demand is deterministic, developing SLTVSP and the proposed approach to cope with the stochastic stations' demands can be another interesting research direction. Moreover, developing new features for the proposed algorithm (e.g., solution representation, crossover, mutation, etc.) and investigating their effects on the results can be further studied. Finally, the performance of other meta-heuristics can be compared with the proposed GA-VNS in this study when addressing the SLTVSP.

Acknowledgements

The authors would like to thank the experts in the real-world case for providing insights into the problem.

Masood Fathi acknowledges the financial support received from the KK-stiftelsen (Knowledge Foundation, Stockholm, Sweden) for the ProSpekt 2018 project OPTION.

Data references

The data that support this study are openly available in [Mendeley Data] at [http://doi.org/\[10.17632/2vd7mpx94m.1\]](http://doi.org/[10.17632/2vd7mpx94m.1]).

References

- Alnahhal, M., & Noche, B. (2015). A genetic algorithm for supermarket location problem. *Assembly Automation*, 35(1), 122–127.
- Battini, D., Gamberi, M., Persona, A., & Sgarbossa, F. (2015). Part-feeding with supermarket in assembly systems: transportation mode selection model and multi-scenario analysis. *Assembly Automation*, 35(1), 149–159.
- Battini, Daria, Boysen, N., & Emde, S. (2013). Just-in-Time supermarkets for part supply in the automobile industry. *Journal of Management Control*, 24(2), 209–217.
- Battini, Daria, Faccio, M., Persona, A., & Sgarbossa, F. (2010). “supermarket warehouses”: Stocking policies optimization in an assembly-to-order environment. *International Journal of Advanced Manufacturing Technology*, 50(5–8), 775–788.
- Belassiria, I., Mazouzi, M., ELfezazi, S., Cherrafi, A., & ELMaskaoui, Z. (2018). An integrated model for assembly line re-balancing problem. *International Journal of Production Research*, 56(16), 5324–5344.
- Boysen, N., Emde, S., Hoeck, M., & Kauderer, M. (2015). Part logistics in the automotive industry: Decision problems, literature review and research agenda. *European Journal of Operational Research*, 242(1), 107–120.
- Boysen, N., & Fliedner, M. (2010). Cross dock scheduling: Classification, literature review and research agenda. *Omega*, 38(6), 413–422.
- Dolgui, A., Tiwari, M. K., Sinjana, Y., Kumar, S. K., & Son, Y. J. (2018). Optimising integrated inventory policy for perishable items in a multi-stage supply chain. *International Journal of Production Research*, 56(1–2), 902–925.
- Emde, S., Abedinnia, H., & Glock, C. H. (2018). Scheduling electric vehicles making milk-runs for just-in-time delivery. *IIE Transactions*, 50(11), 1013–1025.
- Emde, S., & Boysen, N. (2011). Optimally routing and scheduling tow trains for JIT-supply of mixed-model assembly lines. *European Journal of Operational Research*, 217(2), 287–299.
- Emde, S., & Boysen, N. (2012). Optimally locating in-house logistics areas to facilitate JIT-supply of mixed-model assembly lines. *International Journal of Production Economics*, 135(1), 393–402.
- Emde, S., Fliedner, M., & Boysen, N. (2012). Optimally loading tow trains for just-in-time supply of mixed-model assembly lines. *IIE Transactions (Institute of Industrial Engineers)*, 44(2), 121–135.
- Emde, S., & Gendreau, M. (2017). Scheduling in-house transport vehicles to feed parts to automotive assembly lines. *European Journal of Operational Research*, 260(1), 255–267.
- Fathi, M., Alvarez, M. J., Hassani Mehraban, F., & Rodríguez, V. (2014). A Multiobjective optimization algorithm to solve the part feeding problem in mixed-model assembly lines. *Mathematical Problems in Engineering*, 2014, 1–12.
- Fathi, M., Álvarez, M. J., & Rodríguez, V. (2016). A new heuristic-based bi-objective simulated annealing method for U-shaped assembly line balancing. *European Journal of Industrial Engineering*, 10(2), 145–169.
- Fathi, M., Nourmohammadi, A., Ng, A. H. C., Syberfeldt, A., & Eskandari, H. (2019). An improved genetic algorithm with variable neighborhood search to solve the assembly line balancing problem. *Engineering Computations*, In press.

- Fathi, M., Rodríguez, V., & Alvarez, M. J. (2014). A novel memetic ant colony optimization-based heuristic algorithm for solving the assembly line part feeding problem. *International Journal of Advanced Manufacturing Technology*, 75(1–4), 629–643.
- Fathi, M., Rodríguez, V., Fontes, D. B. M. M., & Alvarez, M. J. (2016). A modified particle swarm optimisation algorithm to solve the part feeding problem at assembly lines. *International Journal of Production Research*, 54(3), 878–893.
- Geissdoerfer, M., Morioka, S. N., de Carvalho, M. M., & Evans, S. (2018). Business models and supply chains for the circular economy. *Journal of Cleaner Production*, 190, 712–721.
- Golz, J., Gujjula, R., Günther, H. O., Rinderer, S., & Ziegler, M. (2012). Part feeding at high-variant mixed-model assembly lines. *Flexible Services and Manufacturing Journal*, 24(2), 119–141.
- Liu, X., Lu, S., Pei, J., & Pardalos, P. M. (2018). A hybrid VNS-HS algorithm for a supply chain scheduling problem with deteriorating jobs. *International Journal of Production Research*, 56(17), 5758–5775.
- Lopes de Sousa Jabbour, A. B., Jabbour, C. J. C., Godinho Filho, M., & Roubaud, D. (2018). Industry 4.0 and the circular economy: a proposed research agenda and original roadmap for sustainable operations. *Annals of Operations Research*, 270(1–2), 273–286.
- Mladenovic, N., & Hansen, E. (1997). Variable Neighborhood Search. *Computer.* Ops Res*, 24(1), 1097–1100.
- Nourmohammadi, A., Eskandari, H., & Fathi, M. (2019). Design of stochastic assembly lines considering line balancing and part feeding with supermarkets. *Engineering Optimization*, 51(1), 63–83.
- Osaba, E., Yang, X. S., Diaz, F., Lopez-Garcia, P., & Carballedo, R. (2016). An improved discrete bat algorithm for symmetric and asymmetric Traveling Salesman Problems. *Engineering Applications of Artificial Intelligence*, 48, 59–71.
- Schmid, N. A., & Limère, V. (2019). A classification of tactical assembly line feeding problems. *International Journal of Production Research*, 1–24.
- Taguchi, G., Chowdhury, S., & Wu, Y. (2007). *Taguchi's Quality Engineering Handbook*.
- Urbinati, A., Chiaroni, D., & Chiesa, V. (2017). Towards a new taxonomy of circular economy business models. *Journal of Cleaner Production*, 168, 487–498.
- van Buren, N., Demmers, M., van der Heijden, R., & Witlox, F. (2016). Towards a circular economy: The role of Dutch logistics industries and governments. *Sustainability (Switzerland)*, 8(7).
- Zhao, F., Liu, Y., Zhang, Y., Ma, W., & Zhang, C. (2017). A hybrid harmony search algorithm with efficient job sequence scheme and variable neighborhood search for the permutation flow shop scheduling problems. *Engineering Applications of Artificial Intelligence*, 65, 178–199.
- Zhou, B.-H., & Tan, F. (2019). A self-adaptive estimation of distribution algorithm with differential evolution strategy for supermarket location problem. *Neural Computing and Applications*.
- Zou, X., Liu, L., Li, K., & Li, W. (2018). A coordinated algorithm for integrated production scheduling and vehicle routing problem. *International Journal of Production Research*, 56(15), 5005–5024.

Appendices

Table A1. Transport vehicles' capacity (in bins), shipment and procurement costs (unit cost).




Transport vehicle ID	Transport vehicle type	Sample figure	Capacity	Shipment cost	Procurement cost
1	Trolley		5	3	100
2	Tow train		20	2	500
3	AGV		30	1	1000

Table A2. The stations' demands (in bins) for the case study.

Station #	Demand	Station #	Demand	Station #	Demand
1	10	16	5	31	10
2	6	17	2	32	4
3	7	18	5	33	7
4	4	19	7	34	4
5	5	20	4	35	7
6	4	21	1	36	2
7	10	22	4	37	5
8	6	23	10	38	3
9	3	24	7	39	2
10	3	25	4	40	10
11	9	26	10	41	8
12	2	27	2	42	10
13	8	28	6	43	9
14	9	29	2	44	1
15	8	30	9	45	5