# Extending the solid step fixed-charge transportation problem to consider two-stage networks and multi-item shipments 

Hosna Mollanoori a, Reza Tavakkoli-Moghaddam ${ }^{\text {a,b, }}$, Chefi Triki c,d,*, Mostafa HajiaghaeiKeshteli ${ }^{\text {e }}$, Fatemeh Sabouhi ${ }^{\text {f }}$<br>a School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran ${ }^{b}$ Arts et Métiers ParisTech, LCFC, Metz, France<br>c Department of Engineering for Innovation, University of Salento, Lecce, Italy, ${ }^{d}$ College of Sciences and Engineering, Hamad Bin Khalifa University, Doha-Qatar<br>${ }^{e}$ Department of Industrial Engineering, University of Science and Technology of Mazandaran, Behshahr, Iran f Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran


#### Abstract

This paper develops a new mathematical model for a capacitated solid step fixedcharge transportation problem. The problem is formulated as a two-stage transportation network and considers the option of shipping multiple items from the plants to the distribution centers (DC) and afterwards from DCs to customers. In order to tackle such an NP-hard problem, we propose two meta-heuristic algorithms; namely, Simulated Annealing (SA) and Imperialist Competitive Algorithm (ICA). Contrary to the previous studies, new neighborhood strategies maintaining the feasibility of the problem are developed. Additionally, the Taguchi method is used to tune the parameters of the algorithms. In order to validate and evaluate the performances of the model and algorithms, the results of the proposed SA and ICA are compared. The computational results show that the proposed algorithms provide relatively good solutions in a reasonable amount of time. Furthermore, the related comparison reveals that the ICA generates superior solutions compared to the ones obtained by the SA algorithm.


Keywords: Step fixed-charge transportation; Two-stage; solid; Simulated Annealing; Imperialist competitive algorithm.

## 1. Introduction

Networks have been proved useful in modeling real-world problems and are widely used to represent a variety of systems including transportation and logistics. There are several studies on transportation problems leading to new models and methods. Transportation network design, as one of the most important fields of supply chain management, provides a great opportunity to reduce logistics costs. Hitchcock (1941) presented the first formulation of the classical Transportation Problem (TP) as a network optimization problem. The TP is a practically important and well-known optimization

[^0]problem. It is an integral part of logistics and supply chain management which is critical for decreasing costs and improving service quality (Zhang et al., 2016).

A traditional TP can be stated as a set of sources and destinations with variable transportation costs between each source and destination pair based on the number of transported units. Destinations' demand and capacities of the sources are given. The goal is to determine the number of products transported from each source to each destination in order to minimize the total transportation cost (Sanei et al., 2015). An assumption in traditional transportation problems is to consider the direct cost between two parties according to the number of units transported. However, in most of the real-world applications, particularly in distribution, besides variable transportation costs, a fixed cost for opening each arc is considered which is generally independent of the transported amounts.

A Fixed-Charge Transportation Problem (FCTP) can be applied to many real-world problems such as distribution, production, scheduling, and location. In these cases, fixed costs may include highway charges, setup costs in production systems, landing fees at airports or maintenances costs of roads (Mingozzi and Roberti, 2017). The fixed-charge problem was introduced by Hirsch and Dantzig (1954). Later, Balinski (1961) formulated the fixed-charge transportation problem (FCTP), described its features, and presented the first approximate algorithm for its solution. Several exact solution algorithms were proposed for solving the FCTP including Kowalski et al. (2014) and Adlakha et al. (2010). Due to the fact that an FCTP is a nondeterministic polynomial-time hard (NP-hard) problem (Hirsch and Dantzig, 1954), the computational time to obtain exact solutions increases in an exponential manner. Therefore, as the size of the problem increases, its solution becomes more complicated and time-consuming (Hajiaghaei-Keshteli et al., 2010). To deal with such an NP-hard problem several approximations (e.g., Adlakha and Kowalski (2015)), heuristics (e.g., (Adlakha and Kowalski, 2003, Aguado, 2009)) and metaheuristics (e.g., (Kannan et al., 2014, El-Sherbiny and Alhamali, 2013, Pramanik et al., 2015, Sadeghi-Moghaddam et al., 2017)) have been developed.

Over the years, several extensions have been proposed in order to involve numerous concepts and assumptions including solid FCTP (e.g. (Giri et al., 2015)), multi-stage FCTP ((Calvete et al., 2016, Shirazi et al., 2015), multi-item FCTP (e.g. (Khurana and Adlakha, 2015, Fakhri and Ghatee, 2016, Munguía et al., 2017)), multi-objective (Chen et al., 2017, Khurana and Adlakha, 2015)and step FCTP (e.g.(Sanei et al., 2015, Rajabi et al., 2013, Molla-Alizadeh-Zavardehi et al., 2014a, Molla-Alizadeh-Zavardehi et al., 2014b, ElSherbiny, 2012)).

A solid transportation problem is an extension of the FCTP, in which different types of conveyances (e.g., trucks, trains, ships and cargo flights) participate in the shipping products. As the conveyances capacity is a critical parameter in transportation problems, capacity restrictions are taken into account, besides the source and demand constraints in solid TP. In other words, if only one single type of conveyance is considered, then the problem will convert to the traditional TP. To solve the solid FCTP problem, Ojha et al. (2010) presented a genetic algorithm for a multi-objective and multi-commodity variant of the problem with fuzzy resource and demand as well as considering discount in
transportation costs. In addition, Yang and Liu (2007) studied the solid FCTP problem under a fuzzy environment. They designed a hybrid intelligent algorithm, using fuzzy simulation and a Tabu Search (TS) algorithm, to solve the solid FCTP. They presented three decision models with several criteria and provided the solution of two small sized coal transportation problems. Giri et al. (2015) presented an approach for solving a fully fuzzy solid multi-item FCTP in both balanced and unbalanced types. All parameters and decision variables are considered to be fuzzy in their model. They also investigated and solved the solid FCTP variant of the problem that considers the parameters as uncertain and the decision variables as not fuzzy. One small-sized instance is defined to perform several numerical experiments and the results are compared with the previous studies. Finally, they claimed that their approach works better than the approach presented by Kumar et al. (2011). Later, Gupta et al. (2016) presented a note on paper Giri et al. (2015) stating that their assumptions are not correct and hence their claim is to be rejected. Among the more recent works on the solid FCTP, we can refer to (Chen et al., 2017, Golmohamadi et al., 2017). Chen et al. (2017) studied the uncertain bi-criteria solid FCTP and considered transportation time as a second objective function in addition to the cost criterion. Golmohamadi et al. (2017) employed six new meta-heuristics to solve a fuzzy solid FCTP. The transportation type they considered is based on the batch transformation of products. They compared the results of seven large-scale problems by using their proposed algorithms. Another important aspect of transportation problems is how products are distributed in a supply chain. The shipment of products may be directly from sources to customers or it may be accomplished via distribution centers (DCs). The presence of DCs is modeled as a two-stage supply chain problem. Several studieshave been conducted on FCTP while considering two-stage networks. (e.g. (Ekşioğlu et al., 2007, Jawahar and Balaji, 2009, Panicker et al., 2013)).

A Step Fixed-Charge Transportation Problem (SFCTP) is another variant of FCTP, where the fixed charge is proportional to the amount to be shipped. The SFCTP is known to be an $N P$-super hard problem because of the step function structure of the objective function (Kowalski and Lev, 2008). The SFCTP has not yet been well-probed. It was originally introduced by Sandrock (1988) but the extended formulation of SFCTP has been proposed by Kowalski and Lev (2008). They also presented a simple heuristic algorithm to solve only small-sized instances of the SFCTP. Another approach to tackle this problem was proposed by El-Sherbiny (2012) who developed an alternate Mutation based Artificial Immune (MAI) algorithm and introduced and compared sixteen different mutation functions in MAI algorithm. Unlike Kowalski and Lev (2008), El-Sherbiny's algorithm (2012) is applicable to solve large-scale problems. In addition, Molla-AlizadehZavardehi et al. (2014a) presented a spanning tree based genetic and memetic algorithms to solve the SFCTP. They provided solutions for both small- and large-sized instances. Another study on the Step fixed-charge solid transportation problem was developed by Sanei et al. (2015), who proposed a dual decomposition approach based on Lagrangian relaxation. Their approach has the ability to solve large-scale instances. A Lagrangian heuristic has also been used to produce upper bounds. Table 1 illustrates the features of
the published papers on FCTP which are most relevant to the present research. This table and the literature review on the SFCTP reveal the following gaps:

1) Up to now, there is no any study on the multi-item SFCTP (all available papers have modeled transportation of single product).
2) Single stage is the only network structure that has been employed in SFCTP literature, i.e. involving only direct the transfer mode for the demand satisfaction.
3) None of the studies on SFCTP has considered the capacity restriction which is an important issue in transportation problems. However, in real-world applications, capacity restrictions of the roads enforce vehicles to transport goods with excess capacity.
4) Only one single work has been proposed on solid SFCTP (Sanei et al., 2015).

Table 1
Characteristics of the FCTP in the literature review

| Paper | Solid | Step | Multiitem | Number of stages |  | Capacity constraint |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | one | two |  |
| Yang and Liu (2007) | $\checkmark$ |  |  | $\checkmark$ |  |  |
| Ekşioğlu et al., (2007) |  |  | $\checkmark$ |  | $\checkmark$ | $\checkmark$ |
| Kowalski and Lev (2008) |  | $\checkmark$ |  | $\checkmark$ |  |  |
| Balaji and Jawahar (2010) | $\checkmark$ |  |  |  | $\checkmark$ |  |
| Hajiaghaei-Keshteli et al., (2010) |  |  |  | $\checkmark$ |  |  |
| Ojha et al. (2010) | $\checkmark$ |  |  | $\checkmark$ |  |  |
| El-Sherbiny (2012) |  | $\checkmark$ |  | $\checkmark$ |  |  |
| Panicker et al. (2013) |  |  |  |  | $\checkmark$ |  |
| Rajabi et al. (2013) |  | $\checkmark$ |  | $\checkmark$ |  |  |
| Molla-Alizadeh-Zavardehi et al. (2014b) |  | $\checkmark$ |  | $\checkmark$ |  |  |
| Kannan et al. (2014) |  |  |  |  | $\checkmark$ |  |
| Molla-Alizadeh-Zavardehi et al. (2014a) |  | $\checkmark$ |  | $\checkmark$ |  |  |
| Giri et al. (2015) | $\checkmark$ |  | $\checkmark$ | $\checkmark$ |  |  |
| Thiongane et al. (2015) |  |  | $\checkmark$ | $\checkmark$ |  | $\checkmark$ |
| Pramanik et al. (2015) |  |  |  |  | $\checkmark$ | $\checkmark$ |
| Sanei et al. (2015) | $\checkmark$ | $\checkmark$ |  | $\checkmark$ |  |  |
| Fakhri and Ghatee (2016) |  |  | $\checkmark$ | $\checkmark$ |  | $\checkmark$ |
| Calvete et al. (2016) |  |  |  |  | $\checkmark$ |  |
| Munguía et al. (2017) |  |  | $\checkmark$ | $\checkmark$ |  |  |
| Chen et al. (2017) | $\checkmark$ |  |  | $\checkmark$ |  |  |
| This study | $\checkmark$ | $\checkmark$ | $\checkmark$ |  | $\checkmark$ | $\checkmark$ |

In order to fill the aforementioned gaps and to get closer to real-world applications, we formulate and solve a new variant of the SFCTP as a two-stage supply chain problem in which multiple commodities are shipped from sources (plants) to destinations
(customers) via a set of potential depots (as distribution centers). Note that several conveyances with different capacity can be assigned for transportation in every stage. The aim of the proposed model is to minimize the total transportation costs including variable costs and two types of fixed charges. In other words, a two-stage, multi-item, solid, capacitated SFCTP problem is investigated in this paper. In order to find a good quality solution for the problem, we use both heuristics and metaheuristics. Another contribution of this paper is developing new neighborhood procedures for the metaheuristics.

The rest of the paper is organized as follows: section 2 describes the problem definition and the mathematical model while sections 3 presents the solution procedures via Simulated Annealing (SA) and Imperialist Competitive Algorithm (ICA), respectively. In Section 4, numerical experimental results are provided to assess the efficiency of the proposed algorithms, and section 5 presents the sensitivity analysis. Finally, Section 6 concludes the paper and recommends some areas for further research.

## 2. Problem definition and mathematical model

In this section, we model a multi-item, capacitated solid step FCTP. A two-stage supply chain, including sources, distribution centers and customers (destinations) is also considered. Finding suitable connections (or arcs) between facilities as well as the quantity of shipment on each arc, while meeting the demand, source, and capacity constraints are the aims of the problem. The objective function is the minimization of the transportation costs that consist of two types: a variable cost which is unit-based (i.e., proportional to the number of transported units) and the fixed charge that is independent of the amount of commodity to be transported. The fixed charge will be paid if any nonzero amount of commodity is transported between a source and a destination. Because of the stepwise manner of the problem, there are two types of fixed charges for each stage in the model. When the amount of product is less than a certain threshold (denoted as $A_{i j l}$ for the first stage and $\bar{A}_{j k l}$ for the second stage) the first fixed charge type is applied and when the amount transported exceeds the step parameter the extra fixed charge will be added to the fixed transportation cost. Extra fixed charge happens within the transportation sector when different technologies are available for the shipping and larger amount of products will require the use of a more expensive technology. This also happens when the supply chain uses the auction paradigm for procuring its transportation need and the shipping cost will, thus, increase in a step-wise manner as more conveyances characterized by a higher bidding cost are needed.

A schematic representation of the two-stage, multi-item, solid FCTP is shown in Fig. 1. Possible connections are represented by arrays and the aim is to find the links leading to the minimum total costs.


Fig. 1. Structure of the two-stage FCTP supply chain

### 2.1 Notations

The model sets and indices are as follows:

I Set of sources, indexed by $i \in I$
$J \quad$ Set of distribution centers, indexed by $j \in J$
$K \quad$ Set of customers, indexed by $k \in K$
$L \quad$ Set of conveyances, indexed by $l \in L$
$P \quad$ Set of items, indexed by $p \in P$
The model's parameters are as follows:
$d_{k p} \quad$ Demand at customer $k$ for item $p$
$s_{i p} \quad$ Supply at source $i$ for item $p$
$\operatorname{cap}_{l} \quad$ Capacity of conveyance $l$
$r_{i j l} \quad$ Capacity of arc $(i, j)$ for conveyance $l$
$\bar{r}_{j k l} \quad$ Capacity of arc $(j, k)$ for conveyance $l$
$c_{i j l p} \quad$ Shipping cost per unit transported from source $i$ to $\mathrm{DC} j$ by conveyance $l$ for item $p$
$\bar{c}_{j k l p} \quad$ Shipping cost per unit transported from DC $j$ to customer $k$ by conveyance $l$ for item $p$
$f_{i j l, 1} \quad$ Fixed-charge from source $i$ to $\mathrm{DC} j$ by conveyance $l$ for item $p$
$f_{j k l, 1} \quad$ Fixed-charge from DC $j$ to customer $k$ by conveyance $l$ for item $p$
$f_{i j l, 2} \quad$ Extra fixed-charge from source $i$ to $\mathrm{DC} j$ by conveyance $l$ for item $p$
$f_{j k l, 2}$ Extra fixed-charge from DC j to customer k by conveyance l for item p
$A_{j k l} \quad$ Amount of products for the first stage
$\bar{A}_{j k l} \quad$ Amount of products for the second stage

In addition to some auxiliary state variables that we will introduce within the model, our decision variables are as follows:
$x_{i j l p} \quad$ Amount of item $p$ transported from source $i$ to $\operatorname{DC} j$ by conveyance $l$
$\bar{x}_{j k l p} \quad$ Amount of item $p$ transported from DC $j$ to customer $k$ by conveyance $l$
$y_{i j, 1} \quad$ A binary variable that takes value 1 if any product is transported between source $i$ to DC $j$ by conveyance $l$
$y_{i j, 2}$ A binary variable that takes value 1 if more than $\mathrm{A}_{i j l}$ is transported between source $i$ to $\operatorname{DC} j$ by conveyance $l$
$y_{j k l, 1}$
A binary variable that takes value 1 if any product is transported between DC $j$ to customer $k$ by conveyance $l$
$y_{j k l, 2}$ A binary variable that takes value 1 if more than $\overline{\mathrm{A}}_{j k l}$ is transported between DC $j$ to customer $k$ by conveyance $l$

### 2.2 Model Formulation

$$
\begin{align*}
\operatorname{Min} Z= & \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{l=1}^{L} \sum_{p=1}^{P}\left\{c_{i j p} x_{i j l p}\right\}+\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{l=1}^{L}\left\{f_{i j l, 1} y_{i j l, 1}+f_{i j l, 2} y_{i j l, 2}\right\}+  \tag{1}\\
& \sum_{j=1}^{J} \sum_{k=1}^{K} \sum_{l=1}^{L} \sum_{p=1}^{P}\left\{\bar{c}_{j k l p} \bar{x}_{j k l p}\right\}+\sum_{j=1}^{J} \sum_{k=1}^{K} \sum_{l=1}^{L}\left\{f_{j k l, 1} y_{j k l, 1}+f_{j k l, 2} y_{j k l, 2}\right\}
\end{align*}
$$

s.t.

$$
\begin{align*}
& \sum_{j=1}^{J} \sum_{l=1}^{L} x_{i j p} \leq s_{i p} \quad \forall i \in I, \forall p \in P  \tag{2}\\
& \sum_{j=1}^{J} \sum_{l=1}^{L} \bar{x}_{j k l p}=d_{k p} \quad \forall k \in K, \forall p \in P  \tag{3}\\
& \sum_{i=1}^{I} \sum_{l=1}^{L} x_{i j l p}=\sum_{k=1}^{K} \sum_{l}^{L} \bar{x}_{j k l p} \quad \forall j \in J, \forall p \in P  \tag{4}\\
& \sum_{j=1}^{J} \sum_{k=1}^{K} \bar{x}_{j k l p} \leq \text { cap }_{l} \quad \forall l \in L, p \in P  \tag{6}\\
& \sum_{p=1}^{P} x_{i j l p} \leq r_{i j l} \quad \forall i \in I, \forall \mathrm{j} \in J, \forall \mathrm{l} \in L  \tag{7}\\
& \sum_{p=1}^{P} \bar{x}_{j k l p} \leq \bar{r}_{j k l} \quad \forall j \in J, \forall \mathrm{k} \in K, \forall \mathrm{l} \in L  \tag{8}\\
& x_{i j p} \geq 0 \quad \forall i \in I, \forall j \in J, \forall l \in L, \forall p \in P  \tag{9}\\
& \bar{x}_{j k l p} \geq 0 \quad \forall j \in J, \forall \mathrm{k} \in K, \forall l \in L, \forall p \in P  \tag{10}\\
& y_{i j l, 1}=\left\{\begin{array}{ll}
1 & \text { for } \sum_{p=1}^{P} x_{i j l p}>0 \\
0 & \text { otherwise }
\end{array} \quad \forall i \in I, \forall j \in J, \forall l \in L\right.
\end{align*}
$$

$$
\begin{align*}
& y_{i j l, 2}=\left\{\begin{array}{ll}
1 & \text { for } \sum_{p=1}^{P} x_{i j p} \geq \mathrm{A}_{i j l} \\
0 & \text { otherwise }
\end{array} \quad \forall i \in I, \forall j \in J, \forall l \in L\right.  \tag{12}\\
& y_{j k l, 1}=\left\{\begin{array}{ll}
1 & \text { for } \sum_{p=1}^{P} \bar{x}_{j k l p}>0 \\
0 & \text { otherwise }
\end{array} \quad \forall i \in I, \forall j \in J, \forall l \in L\right. \tag{13}
\end{align*}
$$

The objective function (1) is the step cost function and minimizes the total variable costs and fixed charges. Constraint (2) is the supply constraint and it guarantees that the total quantity shipped to distribution centers does not exceed the source's capacity. Similarly, the demand satisfaction is represented by Constraint (3) stating that the quantity shipped from distribution centers to customers should be equal to the customer demand. Constraint (4) represents the flow conservation showing that the total quantity shipped from sources to distribution centers is equal to the quantity shipped from distribution centers to customers. Constraints (5) and (6) imply that the amount of products shipped with each conveyance does not exceed its capacity. Constraints (7) and (8) are arcs capacity constraints. Constraints (9) and (10) represent the non-negativity of decision variables. Constraints set (11) assigns a value of 1 to $y_{i j l, 1}$ if the product is distributed from source $i$ to distribution center $j$ via conveyance l. Constraint (12) ensures that $y_{i j l, 2}$ equals to 1 if the number of products moved between source $i$ and distribution center $j$ is more than $\mathrm{A}_{i j l}$. Constraint (13) enforces variable $y_{j k l, 1}$ to be 1 if the product is moved from distribution center $j$ to customer $k$ via conveyance l. Constraint (14) ensures that $y_{j k l, 2}$ equals to 1 if the number of products distributed between center $j$ and customer $k$ is more than $\overline{\mathrm{A}}_{j k l}$.

Due to the step structure of objective function, the binary variables are dependent to integer variables. Hence, we define the following auxiliary constraints.

$$
\begin{array}{ll}
\sum_{p=1}^{P} x_{i j l p} \leq r_{i j l} y_{i j l, 1} & \forall i \in I, \forall j \in J, \forall l \in L \\
\sum_{p=1}^{P} \bar{x}_{j k l p} \leq \bar{r}_{j k l} y_{j l l, 1} & \forall j \in J, \forall k \in K, \forall l \in L \\
\sum_{p=1}^{P} x_{i j l p} \leq r_{i j l} y_{i j l, 2}+\mathrm{A}_{i j l}-1 & \forall i \in I, \forall j \in J, \forall l \in L \\
\sum_{p=1}^{P} \bar{x}_{j k l p} \leq \bar{r}_{j l l} y_{j k l, 2}+\overline{\mathrm{A}}_{j k l}-1 & \forall j \in J, \forall k \in K, \forall l \in L
\end{array}
$$

$$
\begin{array}{ll}
\sum_{p=1}^{P} x_{i j l p} \geq \mathrm{A}_{i j l} y_{j l l, 2} & \forall i \in I, \forall j \in J, \forall l \in L \\
\sum_{p=1}^{P} \bar{x}_{j k l p} \geq \overline{\mathrm{A}}_{j k l l} y_{j k l, 2} & \forall j \in J, \forall k \in K, \forall l \in L \tag{20}
\end{array}
$$

Based on the above linearization, our revised model can be summarized as follows:

## Objective Function: (1)

Constraints: (2)-(10) and (15) -(20)

## 3 Proposed SA Approach

As discussed in previous sections, the step FCTP is a very complex class of the NP-hard problems. Its complexity is mainly originated from the step function structure of the objective function (Sandrock, 1988), which necessitates using meta-heuristic algorithms to solve large-scale instances. Therefore, we propose here a SA algorithm (and ICA method in the next section) for solving the proposed model. The SA algorithm is a wellknown metaheuristic algorithm that is inspired by the physical annealing process of solids. In comparison to local search techniques, the main advantages of the SA algorithm are its flexibility and ability to converge to a global solution. Furthermore, the SA algorithm does not depend on the restrictive properties of the model. Hence, it is quite adaptable to different models (Busetti, 2003).

The SA is motivated by the annealing process in the metallurgical industry and it has the capability to escape from being trapped in local optima via accepting non-improving solutions with a certain probability in each temperature. It starts with an initial solution and navigates around it using a variety of neighborhood search structures. This algorithm has been widely applied to numerous complicated, combinatorial, optimization problems in real-life situations (Naderi et al., 2009, Manavizadeh et al., 2013, Mousavi and Tavakkoli-Moghaddam, 2013). Fig. 2 shows the pseudo code of our SA algorithm.

```
Define the cost function and set the parameters.
Set the initial temperature \(T\), the cooling rate \(\alpha\), Max iteration, and Max sub-iteration
Generate an initial seed:
Generate \(X_{j k p}\), using the equivalent cost matrix.
Find four different \(X_{i j l p}\), matrices using \(X_{j k l p}\) and the following four matrices: the equivalent cost
matrix, the variable cost matrix, fixed charge matrix and a random matrix
Find the best \(X_{i j l p}\) (the \(X_{i j l p}\) which has the smallest total cost).
Set BestSol = initial seed.
Compute the objective function ( \(Z\) )
for it \(=1\) :Max iteration
    for subit \(=1\) :Max sub-iteration
Generate neighborhood seed
    Apply mutation operators to \(X_{j k l p}\)
    Find \(X_{i j l p}\) from new \(X_{j k l p}\)
    compute the total costs of new seed (newZ)
Compute Delta \(=\) new \(-Z\)
if Delta < 0
    accept the neighbor seed.
```

```
else
    generate random number r=Uniform[0,1],
    if p=exp[-Delta/T] > r,
        Accept neighbor seed.
        BestSol = neighbor seed.
    end if
end if
end for
T=T\times\alpha
end for
```

Fig. 2. Pseudo code of our SA algorithm

### 3.1 Solution representation

It is important to select an appropriate strategy solution representation considering the direct effect of the representation on the algorithm performance and the quality of results. The matrix representation is a basic representation of transportation problems. The matrix contains the number of products transported between each pair of nodes; hence, the fitness of a solution can be evaluated straightforwardly (Eckert and Gottlieb, 2002). In this paper, we utilize a matrix representation to show the amounts of products shipped in both the stages.

### 3.2 Equivalent cost

We use the approach used by Altassan et al. (2014) to find the equivalent cost for variable and fixed charges. According to this approach, the equivalent cost is achieved by means of expression (21) and it helps to obtain a linear approximation of the problem. We use this equivalent cost to find the initial solution. For an arc between each source and each distribution center and for each conveyance, the corresponding $M_{i j l}$ can be expressed by:

$$
\begin{equation*}
M_{i j l}=\min \left\{\sum_{k=1}^{K} \sum_{p=1}^{P} d_{k p}, \sum_{p=1}^{P} s_{i p}, r_{i j l}, c a p_{l}\right\} \tag{21}
\end{equation*}
$$

The equivalent variable cost is formulated by:
$C F_{i j l}=\left\{\begin{array}{ccc}\frac{f_{i j l, 1}}{M_{i j l}}+c_{i j l} & \text { if } & A_{i j l} \geq M_{i j l} \\ \frac{f_{i j l, 1}+f_{i j l, 2}}{M_{i j l}}+c_{i j l} & \text { if } & A_{i j l}<M_{i j l}\end{array}\right.$
For an arc between each distribution center and each customer and for each conveyance, the corresponding $M_{j k l}$ can be expressed by:
$M_{j k l}=\min \left\{\sum_{p=1}^{P} d_{k p}, r_{j k l}, c a p_{l}\right\}$
Using expression (23), the equivalent cost of the second layer is as follows:

$$
C F_{j k l}= \begin{cases}f_{j k l, 1} / M_{j k l}+c_{j k l} & \text { if } \bar{A}_{j k l} \geq M_{j k l}  \tag{24}\\ \left(f_{j k l, 1}+f_{j k l, 2}\right) / M_{j k l}+c_{j k l} & \text { if } \bar{A}_{j k l}<M_{j k l}\end{cases}
$$

Table 2 gives the values of equivalent variable costs, where $B$ is a very large number.
Table 2. Equivalent variable cost matrix (Yang and Liu, 2007)

|  |  | Distribution center |  |  |  | Customers |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | 1 | 2 | $j$ | $J$ | 1 | 2 | $k$ | K |  |
| Source | 1 | C ${ }_{1}$, | $C F_{1}$ | $C F_{1}$ | $C F_{1}$, | B | $B$ | $B$ | B | $S_{1}$ |
|  | 2 | CF, | $C F_{\text {^ }}$ | $C F_{2}$, | CF, | B | B | B | B | $S$, |
|  | $i$ | CF, | CF, | CFis | CF, | B | $B$ | $B$ | B | $S$; |
|  | $I$ | CF | CF | CF | $C_{\text {F }}$ | B | B | B | B | $S$ |
| Distribution center | 1 | 0 | $B$ | $B$ | $B$ | $C F_{\text {, }}$ <br> CF <br> $C F_{\text {i, }}$ <br> CF | $\begin{aligned} & C F_{1,} \\ & C F_{n} \\ & C F_{i,} \\ & C F_{n} \end{aligned}$ | CF ${ }^{\text {, }}$ <br> CF, <br> $C F_{i k}$ <br> $C F_{n}$ | $\begin{aligned} & C F_{1 v} \\ & C F_{, v} \\ & C F_{i K} \\ & C F_{v v} \end{aligned}$ | $A_{1}$ <br> A, <br> $A$ i <br> A |
|  | 2 | B | 0 | $B$ | B |  |  |  |  |  |
|  | j | $B$ | $B$ | 0 | $B$ |  |  |  |  |  |
|  | $J$ | $B$ | $B$ | B | 0 |  |  |  |  |  |
| Demand |  | $A=\sum D$ | $\sum D$ | $\sum D$ | $\sum D$ | $D_{1}$ | D, | $D_{\downarrow}$ | $D_{r}$ |  |

### 3.3 Generating the initial seed

To generate the initial solutions, we use a heuristic algorithm which is a modification of the Balaji and Jawahar' algorithm (2010) and that can be described as below:

Step 1: The demand values are allocated to $X_{j k l p}$ using the equivalent variable cost matrix.
First, the minimum value of the matrix is found and then its location is chosen for allocation. The amount assigned to this element is the minimum of the relevant values of the demand, the conveyance capacity and the arc capacity. The allocation continues until all demands are satisfied.
Step 2: The number of units shipped from each distribution center is determined by:

$$
\begin{equation*}
A_{j p}=\sum_{k=1}^{K} \sum_{l=1}^{L} \bar{x}_{j k l p} \tag{25}
\end{equation*}
$$

Step 3: To find the values of $X_{i j l p}$, the values of $A_{j p}$ are used. Similar to Step 1, we use $C F_{i j l}$ to find the location for assignment. The minimum of relevant $A_{j p}$, the available source, the available conveyance and arc capacity will be chosen for allocation. This step continues until the values of $A_{j p}$ are completely allocated to $X_{i j l p}$.

### 3.4 Generating the neighborhood seed

To generate the neighborhood seed, some mutation operators are applied to $\bar{X}_{j k l p}$ and then, as mentioned above, $X_{i j l p}$ can be determined using $\bar{X}_{j k l p}$. In this work, we use a unary random mutation that works as follows. For every element of $\bar{X}_{j k l p}$ a random number between 0 and 1 is generated. If this number is less than a certain mutation rate, then that
element is chosen for mutation. We use two mutation operators to find the neighborhood solutions: insertion move and swap mutation. These mutation operators, described below, work in a way that the solution remains feasible.

- Insertion move: In this type of mutation, to transport a part of the amount of a selected element, we choose a new distribution center and a new conveyance. Thus, we select a random distribution center and a random conveyance between the distribution centers and conveyances, respectively. Then we generate a random number between 1 and the amount of selected element. This part will be transported from the new distribution center and by the new conveyance. Fig. 3 shows the insertion mutation.

| $l$ | 1 |  |  |
| ---: | ---: | :---: | :---: |
| $k$ | 1 | 2 | 3 |
| 1 | 261 |  |  |
| 2 |  |  | 77 |
| 3 |  | 185 |  |


$\leadsto$| $l$ |  |  |  |  |  | 1 |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| $\boldsymbol{k}$ | 1 | 2 | 3 |  |  |  |  |  |
| 1 | 261 |  |  |  |  |  |  |  |
| 2 | 80 |  | 77 |  |  |  |  |  |
| 3 |  | 185 |  |  |  |  |  |  |


| $l$ | 2 |  |  |
| :---: | :---: | :---: | :---: |
| $k$ | 1 | 2 | 3 |
| 1 |  |  |  |
| 2 |  | 60 | 55 |
| 3 | 89 |  |  |


| $l$ | 2 |  |  |
| :---: | :---: | :---: | :---: |
|  | 1 | 2 | 3 |
| 1 |  |  |  |
| 2 |  | 60+85 | 55 |
| 3 | 9 |  |  |

Fig. 3. Initial $\bar{X}_{j k l p}$ and its neighborhood seed after applying an insertion move.

## - Swap mutation

We select a random customer and in the relevant column, two elements with different distribution centers and different conveyances are randomly selected and swapped. Fig. 4 shows the suggested swap mutation.

| $l$ | 1 |  |  |
| ---: | :---: | :---: | :---: |
| $k$ | 1 | 2 | 3 |
| $j$ | 1 | 261 |  |
| 1 |  |  |  |
| 2 |  |  | 77 |
| 3 |  | 185 |  |


| $l$ | 1 |  |  |
| :---: | :---: | :---: | :---: |
|  | 1 | 2 | 3 |
| 1 | 89 |  |  |
| 2 |  |  | 77 |
| 3 |  | 185 |  |


| $l$ | 2 |  |  |
| :---: | :---: | :---: | :---: |
| $k$ | 1 | 2 | 3 |
| 1 |  |  |  |
| 2 |  | 60 | 55 |
| 3 | 89 |  |  |

\[

\]

Fig. 4. Initial $\bar{X}_{j k l p}$ and its neighborhood seed after applying swap mutation.

## 4 Proposed ICA Method

The ICA is an evolutionary algorithm introduced by Atashpaz-Gargari and Lucas (2007). ICA is a population-based algorithm which starts with an initial population and
then attempts to eliminate weaker empires in the imperialistic competition process. This will efficiently increase the probability of finding a near-optimal solution and avoiding local optimum (Duan and Huang, 2014). Reliability, accuracy, simplicity, satisfactory convergence speed and time-saving are the most notable advantages of the ICA (Lucas et al., 2010, Nazari-Shirkouhi et al., 2010, Kaveh and Talatahari, 2010, Talatahari et al., 2012, Khabbazi et al., 2009, Lian et al., 2012, Nia et al., 2015). For more details of the ICA, one can refer to Atashpaz-Gargari and Lucas (2007).

The pseudo code of our imperialist competitive algorithm is outlined in Fig. 5.

1. Generate $N_{\text {country }}$ initial countries by using four allocation methods and select $N_{\text {imp }}$ of the best countries to form the imperialists.
2. Apply the double point crossover to improve the colonies.
3. Check the colonies and their relevant costs. In case a colony has more cost compared to its empire, their positions should be changed.
4. Calculate the power for all the imperialists.
5. Find the empire with the worst power and take its weakest colony. Give the selected colony to the empire with the highest chance to own it (Imperialistic competition).
6. Eliminate the empire with no colonies.
7. If only one empire exists, stop otherwise go to Step 2.

Fig. 5. Pseudo code of the ICA.

### 4.1 Generating the initial empires

As mentioned earlier, the ICA as a population-based algorithm starts with an initial population. Hence, firstly, we generate $N_{\text {country }}$ initial countries. The first three countries of the population are generated by means of the equivalent cost matrix, the variable cost matrix and fixed charge matrix, respectively. The remaining countries ( $N_{\text {country }}-3$ ) are generated using a random matrix.

We select $N_{\text {imp }}$ among the best countries to form the imperialists and other countries will be the colonies of those empires. These colonies are divided among the empires depending on the empire power. To perform the division, the normalized cost of each empire is defined by:

$$
\begin{equation*}
C_{n}=\max _{i}\left\{\mathrm{c}_{i}\right\}-c_{n} \tag{26}
\end{equation*}
$$

where $c_{n}$ is the $n$th empire's cost.
Using the normalized cost, the normalized power of each empire will be given by:

$$
\begin{equation*}
\text { power }_{n}=\left|\frac{C_{n}}{\sum_{i=1}^{N_{i n p}} C_{i}}\right| \tag{27}
\end{equation*}
$$

Furthermore, the initial number of colonies for the $\mathrm{n}^{\text {th }}$ imperialist is:

$$
\begin{equation*}
N . \text { Col }_{n}=\operatorname{round}\left\{\text { power }_{n} \cdot\left(N_{\text {col }}\right)\right\} \tag{28}
\end{equation*}
$$

where $N_{\text {col }}$ is the number of colonies. In order to divide the colonies among imperialists, $N . \mathrm{col}_{n}$ colonies are chosen randomly and the $n$-th empire takes them.

### 4.2 Power of empires

The total power of an empire depends on two factors: the power of the imperialist country as well as the power of its colonies. Hence, to define the total power of an empire, the power of the empire country is added to a percentage of the mean power of its colonies.

### 4.3 Movement of colonies toward their imperialists

The movement of colonies toward their empires is the process in which the empires try to improve their colonies. To apply the movement, we use a double point crossover. In this method for each empire and all of its colonies, we generate a random number between zero and one, if this number is less than 0.5 , the crossover will be applied to that colony. For this purpose, one or two columns of the colony are selected randomly and replaced with the corresponding columns of the relevant imperialist.

Despite empires' efforts to perform the above assimilation policy, there are some deviations in the colonies' movements. For this purpose, we consider a deviation rate, then we generate, for every colony, a random number between 0 and 1 . If the random number is lower than the deviation rate, the corresponding colony is regenerated. If the new colony provides a better solution, it will be accepted.

### 4.4 Revolution

The revolution operators in ICA are similar to mutation operators in SA (see 3.2.2). In this stage we generate, for every colony, a random number and revolution occurs for the colonies whose selected number is lower than the revolution rate.

### 4.5 Exchanging the imperialist-colony positions

When a colony moves towards its empire, it may achieve a better performance compared to its corresponding imperialist. This situation happens when the colony's cost becomes lower than the empire's cost. In this case, the positions of the empire and the colony will be exchanged and colonies start moving toward the new empire position.

### 4.6 Imperialistic competition

As mentioned earlier the empires must, in this algorithm, be able to increase their power or maintain their initial power. Hence, in the imperialist competition, each empire, that is not able to increase its power, will lose the competition. During the imperialist competition, the weaker empires will lose their colonies and the more powerful empires will control of these colonies. The empires will start the competition for the worst colonies
of the imperialist, which has the highest total cost and the most powerful empires have higher chances to possess them.

First, the possession probability of each imperialist is calculated based on its total power. To do so, we need the normalized total cost of the $n^{\text {th }}$ empire that is defined by:

$$
\begin{equation*}
N T C_{n}=\max \left\{T C_{n}\right\}-T C_{n} \tag{29}
\end{equation*}
$$

Where $T C_{n}$ is the total cost of the $n$th empire and $N T C_{n}$ denotes the normalized total cost as the representative for the power of the empire. The possession probability of each imperialist is as follows:

$$
\begin{equation*}
P_{p_{n}}=\left|\frac{N T C_{n}}{\sum_{1}^{N_{m p}} N T C_{n}}\right| . \tag{30}
\end{equation*}
$$

Using the possession probabilities, a mechanism similar to the roulette wheel in the genetic algorithm will be used to allocate the weakest colonies to empires. The weaker empires will lose their colonies in the imperialist competition. An empire is eliminated when it loses all of its colonies to other empires. Finally, when all weak empires are eliminated, only one powerful imperialist exists while all other countries are its colonies. In this stage, the algorithm ends.

## 5 Computational experiments

In this section, the performance of the proposed algorithm is evaluated by comparing the results of ICA algorithm to the results provided by SA algorithm and GAMS, as well. To solve exactly the model, we have used GAMS 24.1 .2 on a computer with 2.0 GHz Intel core 2 and 4GB of RAM. First, ten small-sized test instances with different sizes are solved to investigate the efficiency of the proposed algorithm. These instances are benchmarked from an approach proposed in the literature by Hajiaghaei-Keshteli et al., (2010). Table 3 summarizes the values of the parameters, used in such instances.

Table 3
Raw data for parameters.

| parameters | Lower <br> limit | Upper <br> limit | parameters | Lower <br> limit | Upper <br> limit |
| :--- | :---: | :---: | :--- | :---: | :---: |
| Demand | 100 | 130 | Fixed charges | 35 | 75 |
| Supply <br> Conveyance <br> capacity <br> Arc capacity | 120 | 180 | Extra fixed charges | 80 | 100 |

### 5.1. Parameters tuning

To obtain the maximum performance of the algorithm, it is crucial to perform a tuning phase of the algorithms' parameters. We have employed the Taguchi method to tune the parameters due to the high influence of parameter tuning on the performance of the algorithms.

For this aim, we have used the L9 square matrix design with four parameters in three levels in Taguchi (see Table 4). To ensure the accuracy, each experiment is repeated 5 times and the average value of the experiment is considered. In Taguchi technique, Signal-to-Noise ( $\mathrm{S} / \mathrm{N}$ ) ratio measures the variability of the results. The higher values of $\mathrm{S} / \mathrm{N}$ ratio represent smaller variability in the output variables. Therefore, the optimal factor levels are those with the maximum S/N ratios (Naderi et al., 2009). The S/N ratio is defined by:

$$
\begin{equation*}
S / \lambda_{T}=-10 \log _{1 n}\left[\frac{1}{-} \sum^{n} \frac{1}{\sim}\right\rceil \tag{31}
\end{equation*}
$$

where $n$ is the number of repetitions of the experiment in the same conditions, and $y_{i}$ is the experimental result of each repetition.

Table 4
Orthogonal array L9 in the Taguchi method.

| Experiment | A | B | C | D |
| :---: | :---: | :---: | :---: | :---: |
| 1 | 1 | 1 | 1 | 1 |
| 2 | 1 | 2 | 2 | 2 |
| 3 | 1 | 3 | 3 | 3 |
| 4 | 2 | 1 | 2 | 3 |
| 5 | 2 | 2 | 3 | 1 |
| 6 | 2 | 3 | 1 | 2 |
| 7 | 3 | 1 | 3 | 2 |
| 8 | 3 | 2 | 1 | 3 |
| 9 | 3 | 3 | 2 | 1 |

In this work, the following effective parameters are chosen for tuning: mutation rate, maximum sub-iterations number, $\mathrm{T}_{0}$ and $\alpha$ for SA algorithm and $N_{\text {country, }} N_{\text {imp }}$, revolution rate and deviation rate for the imperialist competitive algorithm, respectively. Table 5 lists the selected factors and their corresponding levels.

Table 5
SA and ICA factors and their levels.

| SA |  |  |  | ICA |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Factors | Level 1 | Level <br> 2 | Level <br> 3 | Factors | Level 1 | Level 2 | Level 3 |
| Mutation rate | 0.1 | 0.3 | 0.5 | $N_{\text {country }}$ | 100 | 150 | 200 |
| Sub-iteration | 15 | 20 | 25 | $N_{\text {imp }}$ | 10 | 15 | 20 |
| To | 400 | 500 | 600 | Revolution rate | 0.1 | 0.3 | 0.5 |
| $\alpha$ | 0.9 | 0.95 | 0.98 | Deviation rate | 0.1 | 0.3 | 0.5 |

Fig. 6 shows the mean $\mathrm{S} / \mathrm{N}$ ratio for each level of the SA factors and Fig. 7 shows the mean $\mathrm{S} / \mathrm{N}$ ratio plot for each level of the ICA. According to the obtained results, the optimal factors for SA and ICA are summarized in Table 6.


Fig. 6. Mean $\mathrm{S} / \mathrm{N}$ ratio plot for each level of the SA factors.


Fig. 7. Mean $\mathrm{S} / \mathrm{N}$ ratio plot for each level of the ICA factors.
Table 6
Optimum values of the SA and ICA factors.

| SA |  | ICA |  |  |
| :--- | :---: | :--- | :---: | :---: |
| Factor | Value |  | Factor | Value |
| Mutation rate | 0.1 |  | $N$ | 200 |
| sub-iteration | 20 |  | $N_{\ldots}$ | 20 |
| $T_{-}$ | 400 |  | Revolution rate | 0.1 |
| $\alpha$ | 0.9 |  | Deviation rate | 0.5 |

### 5.2 Experimental results

The quality of the produced solutions is the criteria used to evaluate the efficiency of the algorithms. Considering the difference in the scales of the objective function for each instance, it is not possible to use the objective function values directly. Consequently, we decided to run each problem five times and the relative percentage deviation (RPD) and the best solution error (BSE) are used as the performance criteria for each instance. According to Ruiz and Stützle (2007), Barzinpour et al. (2014) and Sabouhi et al. (2018), the RPD and BSE are obtained as follows:

$$
\begin{equation*}
R P D=\frac{f_{\text {Algorim }}-f_{\text {Best }}}{} \times 100 \tag{32}
\end{equation*}
$$

$B S E=\frac{f_{\text {Best }}-f_{\text {GAMS }}}{} \times 100$
where $f_{\text {Algorithm }}$ is the value of the objective function in each run, $f_{\text {Best }}$ is the best value of objective function among the results of five runs and $f_{\text {GAMS }}$ is the objective function value obtained by using the GAMS software.

Table 7
Characteristics of small-scale instances.

|  | Problem characteristics |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Instance <br> No. | Sources | DC | Customers | Items | Conveyances |
| $\mathbf{1}$ | 1 | 3 | 1 | 1 | 2 |
| $\mathbf{2}$ | 2 | 2 | 1 | 1 | 2 |
| $\mathbf{3}$ | 4 | 2 | 1 | 1 | 1 |
| $\mathbf{4}$ | 3 | 2 | 2 | 2 | 2 |
| $\mathbf{5}$ | 3 | 2 | 3 | 1 | 1 |
| $\mathbf{6}$ | 4 | 3 | 4 | 2 | 2 |
| $\mathbf{7}$ | 5 | 3 | 4 | 3 | 2 |
| $\mathbf{8}$ | 6 | 2 | 2 | 1 | 1 |
| $\mathbf{9}$ | 6 | 2 | 3 | 1 | 2 |
| $\mathbf{1 0}$ | 6 | 3 | 3 | 2 | 2 |
| $\mathbf{1 1}$ | 7 | 3 | 3 | 2 | 3 |
| $\mathbf{1 2}$ | 8 | 3 | 4 | 3 | 4 |

### 5.2.1 Metaheuristics and exact method comparison on small-sized problems

Table 7 shows the characteristics of the small instance. Tables 8 and 9 illustrate, for each problem instance, the computational time, the best and average values of the objective function, RPD, and BSE, for the SA and ICA algorithms, respectively.

Table 8
Comparison of solution values obtained by SA and GAMS software.

| Instance No. | Solution method |  |  |  |  | Average RPD | Average BSE |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Exact solution |  | SA |  |  |  |  |
|  | Objective function | Runtime (sec.) | Avg. objective function | Best objective function | Avg. runtime (sec.) |  |  |
| 1 | 827 | 0.27 | 827 | 827 | 0.12 | 0 | 0 |
| 2 | 1055 | 0.32 | 1055 | 1055 | 0.11 | 0 | 0 |
| 3 | 911 | 0.31 | 911 | 911 | 0.23 | 0 | 0 |
| 4 | 3846 | 0.5 | 3968 | 3900 | 0.32 | 0.02 | 0.01 |
| 5 | 3602 | 0.3 | 3602 | 3602 | 0.13 | 0 | 0 |
| 6 | 8104 | 3 | 8744 | 8743 | 1.22 | 0 | 0.08 |
| 7 | 12300 | 5 | 12731 | 12550 | 4 | 0.01 | 0.02 |
| 8 | 2270 | 0.4 | 2270 | 2270 | 0.07 | 0 | 0 |


| $\mathbf{9}$ | 2717 | 0.3 | 2717 | 2717 | 0.09 | 0 | 0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| $\mathbf{1 0}$ | 5680 | 4 | 5814 | 5799 | 1 | 0.002 | 0.02 |
| $\mathbf{1 1}$ | 6144 | 3 | 6665 | 6665 | 2 | 0 | 0.08 |
| $\mathbf{1 2}$ | 10522 | 65 | 11270 | 11270 | 10 | 0 | 0.07 |
| Average | 4831 | 6.9 | 5088 | 5077 | 1.6 | 0.002 | 0.023 |

Table 9
Comparison of solution values obtained by the ICA and GAMS software.

| Instance No. | Solution method |  |  |  |  | Average RPD | Average BSE |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Exact solution |  | ICA |  |  |  |  |
|  | Objective function | $\begin{aligned} & \text { Runtime } \\ & \text { (sec.) } \end{aligned}$ | Avg. objective function | Best objective function | Avg. runtime (sec.) |  |  |
| 1 | 827 | 0.27 | 827 | 827 | 0.12 | 0 | 0 |
| 2 | 1055 | 0.32 | 1055 | 1055 | 0.20 | 0 | 0 |
| 3 | 911 | 0.31 | 911 | 911 | 0.21 | 0 | 0 |
| 4 | 3846 | 1 | 3846 | 3846 | 1 | 0 | 0 |
| 5 | 3602 | 0.3 | 3602 | 3602 | 0.15 | 0 | 0 |
| 6 | 8104 | 3 | 8153 | 8153 | 1.5 | 0 | 0.006 |
| 7 | 12300 | 5 | 12368 | 12306 | 3 | 0.005 | 0 |
| 8 | 2270 | 0.4 | 2270 | 2270 | 0.14 | 0 | 0 |
| 9 | 2717 | 0.3 | 2717 | 2717 | 0.19 | 0 | 0 |
| 10 | 5680 | 4 | 5713 | 5713 | 2.5 | 0 | 0.006 |
| 11 | 6144 | 6 | 6144 | 6144 | 2 | 0 | 0 |
| 12 | 10522 | 65 | 10900 | 10857 | 21 | 0.003 | 0.04 |
| Average | 827 | 0.27 | 827 | 827 | 0.12 | 0 | 0 |

### 5.2.2 Proposed Metaheuristics comparison on large-scale problems

In this section, we aim to compare the result obtained by the proposed algorithms while solving large-scale instances. To make such a comparison between SA and ICA, the percent reduction of the best solution (\% PRBS) and the percent reduction of the average solution (\% PRAS) are used (Barzinpour et al., 2014).

$$
\begin{align*}
& \% P R B S=\frac{\text { Best }_{S A}-\text { Best }_{I C A}}{} \times 100  \tag{34}\\
& \% P R A S=\frac{\text { Average }_{S A}-\text { Average }_{I C A}}{} \times 100 \tag{35}
\end{align*}
$$

Table 10 shows the characteristics of the large-scale problems used in this experiment and Table 11 presents the numerical results of 12 instances solved by both proposed algorithms.

Figs. 8 and 9 compare the SA and ICA algorithms in terms of the RPD and BSE. As can be seen, the ICA has a superior performance compared to the SA algorithm in both criteria.

Table 10
Characteristics of large-scale instances.

| Instance <br> No. | Problem characteristics |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | Sources | DC | Customers | Items | Conveyances |
| $\mathbf{1 3}$ | 9 | 4 | 5 | 3 | 4 |
| $\mathbf{1 4}$ | 9 | 4 | 7 | 4 | 4 |
| $\mathbf{1 5}$ | 10 | 5 | 8 | 5 | 5 |
| $\mathbf{1 6}$ | 15 | 6 | 11 | 6 | 5 |
| $\mathbf{1 7}$ | 20 | 10 | 11 | 6 | 6 |
| $\mathbf{1 8}$ | 22 | 10 | 15 | 6 | 6 |
| $\mathbf{1 9}$ | 25 | 10 | 15 | 6 | 7 |
| $\mathbf{2 0}$ | 30 | 12 | 18 | 8 | 7 |
| $\mathbf{2 1}$ | 35 | 12 | 20 | 9 | 9 |
| $\mathbf{2 2}$ | 40 | 15 | 20 | 10 | 10 |
| $\mathbf{2 3}$ | 50 | 10 | 50 | 5 | 5 |
| $\mathbf{2 4}$ | 60 | 15 | 60 | 5 | 5 |

Table 11
Comparison of solution values obtained by SA and ICA.

|  | Solution method |  |  |  |  |  | ơ000 | $\begin{aligned} & \frac{0}{3} \\ & \frac{\pi}{3} \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | SA |  |  | ICA |  |  |  |  |
|  | Avg. of the objective function | Min. objective function | Avg. runtime | Avg. of the objective function | Min. objective function | Avg. runtime |  |  |
| 13 | 16804 | 16742 | 176 | 16345 | 16286 | 183 | 2.72 | 2.73 |
| 14 | 32301 | 32066 | 187 | 29842 | 29842 | 196 | 6.94 | 7.61 |
| 15 | 45898 | 45804 | 347 | 45682 | 45589 | 363 | 0.47 | 0.47 |
| 16 | 74296 | 74296 | 695 | 73550 | 72567 | 734 | 2.33 | 1 |
| 17 | 72269 | 72269 | 813 | 71414 | 71414 | 920 | 1.18 | 1.18 |
| 18 | 100960 | 100440 | 1350 | 101800 | 100830 | 1348 | -0.39 | -0.83 |
| 19 | 103680 | 102670 | 1373 | 101540 | 100740 | 1406 | 1.88 | 2.06 |
| 20 | 159594 | 159350 | 1697 | 154040 | 152320 | 1710 | 4.41 | 3.48 |
| 21 | 209320 | 208060 | 1862 | 189500 | 189500 | 2330 | 8.92 | 9.47 |
| 22 | 247050 | 244270 | 2560 | 225146 | 225146 | 2670 | 7.83 | 8.87 |
| 23 | 287520 | 287520 | 2080 | 275998 | 275998 | 2250 | 4 | 4 |
| 24 | 325340 | 324190 | 2260 | 325480 | 325480 | 2330 | -0.4 | -0.04 |
| Avg. | 139586 | 138973 | 1228 | 134195 | 133809 | 1561 | 4.07 | 3.33 |



Fig. 8. Comparison of the RPD between the SA and ICA algorithm


Fig. 9. Comparison of the BSE between the SA and ICA algorithm

Fig. 10 compares the execution times of different problems in the exact solution and the two meta-heuristic algorithms. As shown in this chart, the execution time of GAMS increases exponentially as the problem size increases while its effect on the computational time of the meta-heuristic algorithms is not so significant.


Fig. 10. Comparison of runtime between GAMS software, SA and ICA algorithms

Finally, we compared the convergence of two proposed algorithms, and the results related to instance 7 are depicted in Fig. 11. Running the algorithms for 400 iterations, the best objective value obtained by SA is 12550 while, the ICA has a better performance achieving 12306 for the final objective function.


Fig. 11. Comparing the SA and ICA on instance7

## 6 Sensitivity analysis

In this section, different sensitivity analyses are performed on key parameters to evaluate their impact on the transportation costs. We consider four groups of input parameters to perform the sensitivity analyses: arc capacity, conveyance capacity, step
related parameter and supply parameter. The sensitivity analyses are conducted on instances 7, 10 and 12 to evaluate the impact of adjustments in important parameters on the total costs.

### 6.1 Analysis of the impact of the conveyances and arc capacities

The conveyances capacity and arcs capacity are two influential factors in transportation problems. In this subsection, we investigate the impact of changing conveyances capacity and arcs capacity on the total transportation costs, independently and simultaneously. We used the same instances 7,10 and 12 to examine whether adjustments in conveyances and arcs capacity can be used as a strategy to improve the transportation costs. Fig. 12 illustrates the changes in transportation cost over a range of conveyance capacity and arc capacity levels. A general insight that can be observed is that transportation cost increases with the reduction in both conveyance and arc capacities. Regardless of the arc capacity variations, with an increase in conveyance capacity, the transportation cost will increase. This can be explained in this way: the more increase in conveyance capacities, the more items with less costly conveyances can be transported. In instance 10, for example, if the capacity of each conveyance increases by $15 \%$, the total costs decrease by $0.86 \%$. In instances 7 and 10 , more than $20 \%$ reduction in conveyance capacity would make the model infeasible. For instance 12 , this threshold is $30 \%$.

Moreover, regardless of conveyance capacity changes, the total costs decrease as the capacity of arcs expands. This occurs because higher arc capacities provide an opportunity to decide to open arcs at less cost. In these cases, the maximum reduction in arc capacity that will maintain the feasibility of the model is $30 \%$. Furthermore, in instance 7 , for example, up to $25 \%$ increase in conveyance capacity and up to $20 \%$ increase in arc capacity lead to a reduction in the total costs and further increase has no effect on the transportation cost. It is worth noting that the changes in the transportation cost value are not proportional to the problem size. In instances 7 and 10, the model is more sensitive to the conveyance capacity variations than to the arc capacity, but in instance 12 the slope of arc capacity curve is steeper than the conveyance curve.

Not surprisingly, Figure 12 shows that the simultaneous changes in capacity parameters lead to more changes in the total costs in comparison to independent changes in arc and conveyance capacities. Considering the above-mentioned explanation, this behavior is logical.


Fig. 12. Impact of the capacity parameters on the transportation cost

### 6.2 Analysis of the impact of the step parameter

The step parameter is one of the key parameters in step FCTP. In this section, we aim to evaluate the effect of this parameter changes on transportation costs.

Fig. 13 illustrates the impact of changes in the step parameter on transportation cost. According to this figure, with the decrease in this parameter; an extra fixed cost is imposed on the problem. Increasing the value of the step parameter provides the opportunity to transport items at a lower cost. As seems to be obvious from this figure, instance 7 is more sensitive to step parameter changes whereas the changes in the step parameter do not have a significant effect on instance 10.


Fig. 13. Impact of the step parameter variations on the transportation cost

### 6.3 Analysis of the impact of the supply

Fig. 14 illustrates how the total cost is affected by the changes in values of supply in the case of three test instances. Obviously, any increase in the supply will result in reduced total cost. All the three instances follow similar patterns. In instance 10, for example, if the available supply increases by $20 \%$, the total cost decreases by $1.47 \%$. However, the curves related to different problems do not have the same slope and the cost-saving value is not proportionate to the problem size. (i.e. instance 12 is less sensitive to the supply changes compared to the other two instances). As it is obvious from Fig. 14, a $20 \%$ increase in the supply will result in a total cost decreases of less than $0.5 \%$ and more than $20 \%$ increase has no significant effect on the total costs.


Fig. 14. Impact of the supply variations on the transportation cost

## 7. Conclusions and future research avenues

In this paper, we have formulated a two-layer, multi-commodity and capacitated solid SFCTP. The proposed model has been solved by two meta-heuristic algorithms, namely SA and ICA. In order to calibrate the parameters of the proposed algorithms, the Taguchi parameter design method has been utilized. We have simulated 12 instances using the SA and ICA algorithms as well as GAMS software to show the efficiency of the proposed algorithms. The performance of the SA and ICA algorithms has confirmed that the algorithms can be considered as viable solutions to cope with SFCTPs. Furthermore, SA and ICA algorithms have been further compared on a set of large-scale instances. The computational results have shown that the ICA is more efficient in comparison to SA. In addition, a sensitivity analyses study is carried out on four parameters, including arc capacity, vehicle capacity, the parameter related to the step structure of the problem and the supplied parameter, and their effect on transportation cost is analyzed. The results show that the transportation cost is inversely proportional to both the aforementioned parameters.

As a direction for future research, more investigation can be carried out to develop a multi-step FCTP and also to include the aspects related to the sustainability in the FCTP as well. In addition, due to the importance of the distribution of perishable goods with the highest possible quality in a today's industry, the transportation of perishable or breakable products can also be taken into consideration in the SFCTP as an additional constraint. Finally, another direction for future investigation can consist in considering the effects of disruptions in transportation links and facilities.

## References

Adlakha, V. \& Kowalski, K. 2003. A Simple Heuristic For Solving Small Fixed-Charge Transportation Problems. Omega, 31, 205-211.
Adlakha, V. \& Kowalski, K. 2015. Fractional Polynomial Bounds For The Fixed Charge Problem. Journal Of Optimization Theory And Applications, 164, 1026-1038.
Adlakha, V., Kowalski, K. \& Lev, B. 2010. A Branching Method For The Fixed Charge Transportation Problem. Omega, 38, 393-397.

Aguado, J. S. 2009. Fixed Charge Transportation Problems: A New Heuristic Approach Based On Lagrangean Relaxation And The Solving Of Core Problems. Annals of Operations Research, 172, 45-69.
Altassan, K. M., El-Sherbiny, M. M. \& Abid, A. D. 2014. Artificial Immune Algorithm For Solving Fixed Charge Transportation Problem. Appl. Math, 8, 751-759.
Atashpaz-Gargari, E. \& Lucas, C. Imperialist Competitive Algorithm: An Algorithm For Optimization Inspired By Imperialistic Competition. 2007 Ieee Congress On Evolutionary Computation, 2007. Ieee, 4661-4667.
Balaji, A. \& Jawahar, N. 2010. A Simulated Annealing Algorithm For A Two-Stage Fixed Charge Distribution Problem Of A Supply Chain. International Journal Of Operational Research, 7, 192-215.
Balinski, M. L. 1961. Fixed-Cost Transportation Problems. Naval Research Logistics Quarterly, 8, 41-54.
Barzinpour, F., Saffarian, M., Makoui, A. \& Teimoury, E. 2014. Metaheuristic Algorithm For Solving Biobjective Possibility Planning Model Of Location-Allocation In Disaster Relief Logistics. Journal Of Applied Mathematics, 2014.
Busetti, F. 2003. Simulated Annealing Overview. Jp Morgan, Italy.
Calvete, H. I., Gale, C. \& Iranzo, J. A. 2016. An Improved Evolutionary Algorithm For The Two-Stage Transportation Problem With Fixed Charge At Depots. Or Spectrum, 38, 189-206.
Chen, L., Peng, J. \& Zhang, B. 2017. Uncertain Goal Programming Models For Bicriteria Solid Transportation Problem. Applied Soft Computing, 51, 49-59.
Duan, H. \& Huang, L. 2014. Imperialist Competitive Algorithm Optimized Artificial Neural Networks For Ucav Global Path Planning. Neurocomputing, 125, 166-171.
Eckert, C. \& Gottlieb, J. Direct Representation And Variation Operators For The Fixed Charge Transportation Problem. International Conference On Parallel Problem Solving From Nature, 2002. Springer, 77-87.
Ekșioğlu, S. D., Ekșioğlu, B. \& Romeijn, H. E. 2007. A Lagrangean Heuristic For Integrated Production And Transportation Planning Problems In A Dynamic, Multi-Item, TwoLayer Supply Chain. Iie Transactions, 39, 191-201.
El-Sherbiny, M. M. 2012. Alternate Mutation Based Artificial Immune Algorithm For Step Fixed Charge Transportation Problem. Egyptian Informatics Journal, 13, 123-134.
El-Sherbiny, M. M. \& Alhamali, R. M. 2013. A Hybrid Particle Swarm Algorithm With Artificial Immune Learning For Solving The Fixed Charge Transportation Problem. Computers \& Industrial Engineering, 64, 610-620.
Fakhri, A. \& Ghatee, M. 2016. Application Of Benders Decomposition Method In Solution Of A Fixed-Charge Multicommodity Network Design Problem Avoiding Congestion. Applied Mathematical Modelling, 40, 6468-6476.
Giri, P. K., Maiti, M. K. \& Maiti, M. 2015. Fully Fuzzy Fixed Charge Multi-Item Solid Transportation Problem. Applied Soft Computing, 27, 77-91.
Golmohamadi, S., Tavakkoli-Moghaddam, R. \& Hajiaghaei-Keshteli, M. 2017. Solving A Fuzzy Fixed Charge Solid Transportation Problem Using Batch Transferring By New Approaches In Meta-Heuristic. Electronic Notes In Discrete Mathematics, 58, 143-150. Gupta, G., Kaur, J. \& Kumar, A. 2016. A Note On "Fully Fuzzy Fixed Charge Multi-Item Solid Transportation Problem". Applied Soft Computing, 41, 418-419.
Hajiaghaei-Keshteli, M., Molla-Alizadeh-Zavardehi, S. \& Tavakkoli-Moghaddam, R. 2010. Addressing A Nonlinear Fixed-Charge Transportation Problem Using A Spanning Tree-Based Genetic Algorithm. Computers \& Industrial Engineering, 59, 259-271.
Hirsch, W. M. \& Dantzig, G. B. 1954. The Fixed Charge Problem. Dtic Document.

Hitchcock, F. L. 1941. The Distribution Of A Product From Several Sources To Numerous Localities. Journal Of Mathematics And Physics, 20, 224-230.
Jawahar, N. \& Balaji, A. 2009. A Genetic Algorithm For The Two-Stage Supply Chain Distribution Problem Associated With A Fixed Charge. European Journal of Operational Research, 194, 496-537.
Kannan, D., Govindan, K. \& Soleimani, H. 2014. Artificial Immune System And Sheep Flock Algorithms For Two-Stage Fixed-Charge Transportation Problem. Optimization, 63, 1465-1479.
Kaveh, A. \& Talatahari, S. 2010. Optimum Design Of Skeletal Structures Using Imperialist Competitive Algorithm. Computers \& Structures, 88, 1220-1229.
Khabbazi, A., Atashpaz-Gargari, E. \& Lucas, C. 2009. Imperialist Competitive Algorithm For Minimum Bit Error Rate Beamforming. International Journal Of Bio-Inspired Computation, 1, 125-133.
Khurana, A. \& Adlakha, V. 2015. On Multi-Index Fixed Charge Bi-Criterion Transportation Problem. Opsearch, 52, 733-745.
Kowalski, K. \& Lev, B. 2008. On Step Fixed-Charge Transportation Problem. Omega, 36, 913-917.
Kowalski, K., Lev, B., Shen, W. \& Tu, Y. 2014. A Fast And Simple Branching Algorithm For Solving Small Scale Fixed-Charge Transportation Problem. Operations Research Perspectives, 1, 1-5.
Kumar, A., Kaur, J. \& Singh, P. 2011. A New Method For Solving Fully Fuzzy Linear Programming Problems. Applied Mathematical Modelling, 35, 817-823.
Lian, K., Zhang, C., Shao, X. \& Gao, L. 2012. Optimization Of Process Planning With Various Flexibilities Using An Imperialist Competitive Algorithm. The International Journal Of Advanced Manufacturing Technology, 59, 815-828.
Lucas, C., Nasiri-Gheidari, Z. \& Tootoonchian, F. 2010. Application Of An Imperialist Competitive Algorithm To The Design Of A Linear Induction Motor. Energy Conversion And Management, 51, 1407-1411.
Manavizadeh, N., Hosseini, N.-S., Rabbani, M. \& Jolai, F. 2013. A Simulated Annealing Algorithm For A Mixed Model Assembly U-Line Balancing Type-I Problem Considering Human Efficiency And Just-In-Time Approach. Computers \& Industrial Engineering, 64, 669-685.
Mingozzi, A. \& Roberti, R. 2017. An Exact Algorithm For The Fixed Charge Transportation Problem Based On Matching Source And Sink Patterns. Transportation Science, 52(2), 229-238.
Molla-Alizadeh-Zavardehi, S., Mahmoodirad, A. \& Rahimian, M. 2014a. Step Fixed Charge Transportation Problems Via Genetic Algorithm. Indian Journal Of Science And Technology, 7, 949-954.
Molla-Alizadeh-Zavardehi, S., Sanei, M., Soltani, R. \& Mahmoodirad, A. 2014b. Solving A Step Fixed Charge Transportation Problem By A Spanning Tree-Based Memetic Algorithm. International Journal Of Mathematical Modelling \& Computations, 4, 177187.

Mousavi, S. M. \& Tavakkoli-Moghaddam, R. 2013. A Hybrid Simulated Annealing Algorithm For Location And Routing Scheduling Problems With Cross-Docking In The Supply Chain. Journal Of Manufacturing Systems, 32, 335-347.
Munguía, L.-M., Ahmed, S., Bader, D. A., Nemhauser, G. L., Goel, V. \& Shao, Y. 2017. A Parallel Local Search Framework For The Fixed-Charge Multicommodity Network Flow Problem. Computers \& Operations Research, 77, 44-57.

Naderi, B., Zandieh, M. \& Fatemi Ghomi, S. 2009. Scheduling Job Shop Problems With Sequence-Dependent Setup Times. International Journal Of Production Research, 47, 5959-5976.
Nazari-Shirkouhi, S., Eivazy, H., Ghodsi, R., Rezaie, K. \& Atashpaz-Gargari, E. 2010. Solving The Integrated Product Mix-Outsourcing Problem Using The Imperialist Competitive Algorithm. Expert Systems With Applications, 37, 7615-7626.
Nia, A. R., Far, M. H. \& Niaki, S. T. A. 2015. A Hybrid Genetic And Imperialist Competitive Algorithm For Green Vendor Managed Inventory Of Multi-Item Multi-Constraint Eoq Model Under Shortage. Applied Soft Computing, 30, 353-364.
Ojha, A., Das, B., Mondal, S. \& Maiti, M. 2010. A Solid Transportation Problem For An Item With Fixed Charge, Vechicle Cost And Price Discounted Varying Charge Using Genetic Algorithm. Applied Soft Computing, 10, 100-110.
Panicker, V. V., Vanga, R. \& Sridharan, R. 2013. Ant Colony Optimisation Algorithm For Distribution-Allocation Problem In A Two-Stage Supply Chain With A Fixed Transportation Charge. International Journal Of Production Research, 51, 698-717.
Pramanik, S., Jana, D. K., Mondal, S. \& Maiti, M. 2015. A Fixed-Charge Transportation Problem In Two-Stage Supply Chain Network In Gaussian Type-2 Fuzzy Environments. Information Sciences, 325, 190-214.
Rajabi, F., Najafi, S., Keshteli, M. H. \& Zavardehi, S. M. A. 2013. Solving Fuzzy Step Fixed Charge Transportation Problems Via Metaheuristics. International Journal Of Research In Industrial Engineering, 2, 24-34.
Ruiz, R. \& Stutzle, T. 2007. A Simple And Effective Iterated Greedy Algorithm For The Permutation Flowshop Scheduling Problem. European Journal Of Operational Research, 177, 2033-2049.
Sabouhi, F., Bozorgi-Amiri, A., Moshref-Javadi, M. \& Heydari, M. 2018. An Integrated Routing And Scheduling Model For Evacuation And Commodity Distribution In LargeScale Disaster Relief Operations: A Case Study. Annals Of Operations Research, 1-35.
Sadeghi-Moghaddam, S., Hajiaghaei-Keshteli, M. \& Mahmoodjanloo, M. 2017. New Approaches In Metaheuristics To Solve The Fixed Charge Transportation Problem In A Fuzzy Environment. Neural Computing And Applications, 1-21.
Sandrock, K. 1988. A Simple Algorithm For Solving Small, Fixed-Charge Transportation Problems. Journal Of The Operational Research Society, 39, 467-475.
Sanei, M., Mahmoodirad, A., Niroomand, S., Jamalian, A. \& Gelareh, S. 2015. Step FixedCharge Solid Transportation Problem: A Lagrangian Relaxation Heuristic Approach. Computational And Applied Mathematics, 36, 1-21.
Shirazi, N., Seyyed Esfahani, M. \& Soleimani, H. 2015. Modeling And Solving A Three-Stage Fixed Charge Transportation Problem Considering Stochastic Demand And Price. Journal Of Industrial Engineering And Management Studies, 2, 27-40.
Talatahari, S., Azar, B. F., Sheikholeslami, R. \& Gandomi, A. 2012. Imperialist Competitive Algorithm Combined With Chaos For Global Optimization. Communications In Nonlinear Science And Numerical Simulation, 17, 1312-1319.
Thiongane, B., Cordeau, J.-F. \& Gendron, B. 2015. Formulations For The Nonbifurcated Hop-Constrained Multicommodity Capacitated Fixed-Charge Network Design Problem. Computers \& Operations Research, 53, 1-8.
Yang, L. \& Liu, L. 2007. Fuzzy Fixed Charge Solid Transportation Problem And Algorithm. Applied Soft Computing, 7, 879-889.
Zhang, B., Peng, J., Li, S. \& Chen, L. 2016. Fixed Charge Solid Transportation Problem In Uncertain Environment And Its Algorithm. Computers \& Industrial Engineering, 102, 186-197.


[^0]:    * Corresponding author.

    E-mail address: chefi.triki@unisalento.it (C. Triki).

