Journal of Optimization in Industrial Engineering
Vol.13, Issue 1, Winter & Spring 2020, 39-53
DOI: 10.22094/JOIE.2017.728.1463

Research Article

Optimization of Multi-period Three-echelon Citrus Supply Chain Problem

Navid Sahebjamnia[a,*], Fariba Goodarzian[a], Mostafa Hajiaghaei-Keshteli[a]

[a] Department of Indusrtial Engineering, University of Science and Technology of Mazandaran, Behshahr, Iran.
Received 07 February 2017; Revised 23 August 2017; Accepted 02 November 2017

Abstract

In this paper, a new multi-objective integer non-linear programming model is developed for designing citrus three-echelon supply chain network. Short harvest period, product specifications, high perished rate, and special storing and distributing conditions make the modeling of citrus supply chain more complicated than other ones. The proposed model aims to minimize network costs including waste cost, transportation cost, and inventory holding cost, and to maximize network's profits. To solve the model, firstly the model is converted to a linear programming model. Then three multi-objective meta-heuristic algorithms are used including MOPSO, MOICA, and NSGA-II for finding efficient solutions. The strengths and weaknesses of MOPSO, MOICA, and NSGA-II for solving the proposed model are discussed. The results of the algorithms have been compared by several criteria consisting of number of Pareto solution, maximum spread, mean ideal distance, and diversification metric. Computational results show that MOPSO algorithm finds competitive solutions in compare with NSGA-II and MOICA.

***Keywords*: Citrus supply chain; Network design; Location- allocation problem; Multi-objective meta-heuristic algorithms

1. Introduction

The Supply Chain Network Design (SCND) problem is an important strategic decision in supply chain management (Pishvaee, 2017 and Hafezalkotob, Khalili-Damghani, & Ghashami, 2016). The SCND problems include all activities of value chain from supply raw material to delivery final products to the customers (Derwik, & Hellström, 2017). Managing facility location, product flow, warehousing, ordering, and distributing are known as the main problems in SCND literature (Christopher, 2016). In addition, the characteristics of the products or services should be considered to model and solve SCND problems. Over the last decades, academics and practitioners of food industries aim to design efficient food supply chain networks at national or global level. To ensure food security and prevent malnutrition, scholars need to develop quantitative models of food supply chain network (Ju, Osako, & Harashina, 2017). Govindan (2018) emphasized that four main activities of food industries including production, processing, transportation, and consumption should be considered in SCND problems. He reviewed food supply chain and identified the main indicators, drivers, and barriers for designing sustainable food supply chain networks. Manders, Caniëls, & Ghijsen (2016) introduced four main organizations of the food supply chain including suppliers, main manufacturer, the logistics service provider, and retailers. They found out the flexibility of each organization through food supply chain network impact on the network directly. conditions led to the differentiation of designing fresh fruit with foods supply chain network (Lowe, & Preckel, 2004). Soto-Silva et al. (2016) reviewed operation research methods in fresh fruit supply chain. They outlined the comprehensive quantitative approaches for designing fresh fruit supply chain as the main gap of the relevant literature. Furthermore, the growing fruit industries scale and a number of autonomous organizations in fruit supply chain were introduced as the main reasons for developing new operation research models in fresh fruit SCND context. However few researches developed operation research methods for designing fresh fruit SCND problems in the literature as noted in (Ahumada, & Villalobos, 2011). For example, González-Araya, Soto-Silva, & Espejo (2015) developed a mixed integer linear programming model to support harvest planning. The model aims to minimize the amount of allocated resources costs (e.g. labor and equipment) and ensure the quality of the fresh fruit along harvest time windows. Nadal-Roig, & Plà-Aragonés (2015) developed a linear programming model for planning daily transport of fresh fruit from the warehouse to processing plants by minimizing transportation costs. Negi and Anand (2015) considered the problems affecting the supply chain of fresh fruits industries in India and suggested appropriate supply chain strategies to overcome the challenges. Bortolini et al. (2016) proposed a three-objective distribution planner to tackle the tactical optimization issue of a fresh food distribution network. The optimization objectives were to minimize operating cost, carbon footprint and delivery time. Nevertheless, the limit research aim to develop operation research models to solve fresh fruit SCND problems. Since the citrus is one

*Corresponding author Email address: n.sahebjamnia@mazust.ac.ir

of the main orchards of the Iran, this study develops a new mathematical model for designing citrus supply chain network. In the proposed model production, processing, and distributing were considered as three main operations of the fresh fruit SCND problem. In this study the operation research model is applied to optimize location of the facility, allocation of products to facilities, and logistic issues.

Melo, Nickel, & Saldanha-Da-Gama (2009) emphasized that SCND problems are complex and meta-heuristic algorithms should be developed to solve large-scale problems. In this regards various meta-heuristic algorithms have been developed to solve large-scale SCND problems. Furthermore, most of the scholars focused on developing multi-objective mathematical programming in order to SCND problems (Harris, Mumford, & Naim 2014). Shanker et al. (2013) developed a multi-objective mathematical model for optimizing total cost and customer demands be met for a single product four echelons SCND. To find the optimum number and location of nodes in network and the flow of product among chain, a hybrid evolutionary based meta-heuristic algorithm has been developed. Saeedi Mehrabad et al. (2017) emphasized that researchers should focus on customer satisfaction factor as well as total costs of SCN in the present competitive market. A multi-objective mathematical programming model was developed for location and allocation problem in the multi-level SCND. To solve the proposed model a hybrid meta-heuristic algorithm has been developed and the results demonstrated the quality of the proposed algorithm. As explored in the literature, different multi-objective meta-heuristic algorithm should be applied to solve large-scale SCND problems such as Multi-Objective Particle Swarm Optimization (MOPSO), Non-dominated Sorting Genetic Algorithm II (NSGA-II), and Multi-Objective Imperialist Competitive Algorithm (MOICA).

MOPSO algorithm as a well-known multi-objective meta-heuristic algorithm that was first introduced by (Eberhart, & Kennedy, 1995) has been applied for solving different problems such as reliability (Khalili-Damghani, Abtahi, & Tavana, 2013), energy management (Litchy, & Nehrir, 2014), layout problem (Ghodratnama, Jolai, & Tavakkoli-Moghaddam, 2015), scheduling (Torabi, et al., 2013), relief chain (Bozorgi-Amiri, et al., 2012), and SCND (Govindan et al., 2014 and Mousavi et al., 2017). Govindan et al. (2014) proposed a novel hybrid multi-objective meta-heuristic algorithm consisting of MOPSO and multi-objective variable neighborhood search. The proposed hybrid MOPSO obtained better solution in comparison with other meta-heuristics. Mousavi et al. [26] developed a MOPSO for finding optimal location of facilities as well as quantity of the order and inventory through supply chain. Their results show that the performance of the MOPSO is better than genetic algorithm in large scale problem. Here, a modified MOPSO algorithm is coded to solve the proposed citrus SCND problem. To demonstrate the efficiency of the modified MOPSO algorithm, two well-known meta-heuristics algorithm including NSGA-II and MOICA are utilized for large scale problems. In addition, the proposed model has been solved with Branch and Bound approach for small size problem and the result show the acceptable gap between and applied methods. Consequently, the main contributions of the research could be highlighted as follows:

- Developing a new mathematical model to design citrus supply chain network;
- Solving the large-scale problem by using three well-known meta-heuristic algorithms including MOPSO, NSGA-II, and MOICA;
- Evaluating the performance of the MOPSO, MOICA, and NSGAII algorithms for finding Pareto solution of citrus SCND problems;
- Designing a new mixed integer programming model to find facility location, flow, and transportation problems of citrus supply chain network.

2. Proposed Model

In this section, a three echelons citrus supply chain network including suppliers, distributors, and customers is described. Gardeners are known as suppliers and placed in the first layer of the network. In the second layer there are distributors who purchase the product from suppliers and distribute among customers in the third level after processing. Product processing involves washing, waxing and sizing operations. Distributors can store product before processing or afterwards. Due to the limited harvesting period, distributors have to purchase products from their suppliers at their capacity and process them along planning horizon. Due to the storage capacity and budget limitation of the distributors in the harvesting period, they can purchase unprocessed products from other distributors after the end of the harvest period. Thus, after product processing, they can be distributed to customers. Due to the fact that the price of the product after the harvest period increases with decreasing supply, distributors prefer to buy and store the product with maximum capacity. Accordingly, the price of supply of the unprocessed product by the distributor will be higher than the supplier. Distributors can use rental depots to increase their capacity and reduce their costs. In this way, they can store more products in the harvesting period. Also, the cost of the warehouse has declined and only the cost of renting the warehouse is added to their costs. To transport the product between the echelons, there are different transportation equipment such as the vans, truck and trailer. Each of the transportation equipment has its own expense and capacity. In this study, suppliers for each distributor would be selected. In addition, the amount of flows in the network between the various echelons, including supplier to the distributor, distributor to the distributor, and distributor to the customer are determined. Finally, the type of transportation equipment is selected for the transportation of the product among different echelons.

2.1. Assumptions

The following assumptions are made in the network configuration:

- Considering the nature of citrus, the eight months planning period was considered.
- The harvest period is set to three months.
- Distributors can only purchase product from suppliers during harvesting (i.e. the first three months).
- The model is designed for a single product, multi-objective, and multi-period SCND problem.
- Distributors can only buy unprocessed products from each other.
- The price of the product will vary in different periods.
- The budget and storage capacity of each distributor is limited and predetermined.
- Processed and unprocessed products are perished in the distributor's warehouse with a different waste rate.

2.2. Notation

The notations including indices, parameters, and, decision variables are:

Sets:

p	sets of supplier ($p = 1,2,\ldots,P$)
d, d'	sets of distributors ($d = 1,2,\ldots,D$)
j	sets of customers ($j = 1,2,\ldots,J$)
t	sets of periods ($t = 1,2,\ldots,T$)
m	Set of transportation equipment ($m = 1,2,\ldots,M$)

Parameters

S_p^{Pt}	Sales price of supplier *p* in period *t*
$S_{dd'}^{Dt}$	Sales price of distributor *d* to distributor *d'* in period *t*
S_{dj}^{Jt}	Sales price of distributor *d* to customer *j* in period *t*
λ_p^{Pt}	Capacity of supplier *p* in period *t*
λ_d^{S}	Unprocessed product storage capacity in warehouse of distributor *d*
λ_d^{F}	Processed product storage capacity in warehouse of distributor *d*
λ_d^{Rt}	Processing capacity of distributor *d* in period *t*
λ_j^{Jt}	Capacity of customer *j* in period *t*
$\lambda_j^{J^+t}$	Capacity upper limit of customer *j* in period *t*
$\lambda_j^{J^-t}$	Capacity lower limit of customer *j* in period *t*
C_{pdm}^{Tt}	Transportation cost from supplier *p* to distributor *d* in period *t* with transportation equipment *m*
$C_{dd'm}^{Tt}$	Transportation cost from distributor *d* to distributor *d'* in period *t* with transportation equipment *m*
C_{djm}^{Tt}	Transportation cost from distributor *d* to customer *j* in period *t* with transportation equipment *m*
C_d^{Rt}	Processing cost of distributor *d* in period *t*
C_d^{S}	Holding cost of unprocessed product for distributor *d*
C_d^{F}	Holding cost of processed product for distributor *d*
C_d^{CSt}	rent cost of unprocessed product for distributor *d* in period *t*
C_d^{CFt}	rent cost of processed product for distributor *d* in period *t*
V_m	Capacity of transportation equipment *m*
B_d^t	Budget of distributor *d* in period *t*
α'_d	Waste rate of unprocessed product for distributor *d*
α_d	Waste rate of processed product for distributor *d*

Decision variables

X_{pd}^{t}	Amount of product be purchased by distributor *d* from supplier *p* in period *t*
$Y_{dd'}^{t}$	Amount of product be purchased by distributor d' from *d* to in period *t*
Z_{dj}^{t}	Amount of product be purchased by customer *j* from distributor *d* in period *t*
W_d^t	Amount of unprocessed product in distributor *d* warehouse in period *t*
K_d^t	Amount of processed product in distributor *d* warehouse in period *t*
U_{pdm}^{t}	Number of transportation equipment *m* needed for transporting product from supplier *p* to distributor *d* in period *t*
$U_{dd'm}^{t}$	Number of transportation equipment *m* needed for transporting product from distributor *d* to *d'* in period *t*
U_{djm}^{t}	Number of transportation equipment *m* needed for transporting product from distributor *d* to customer *j* in period *t*
ω_d^t	Surplus capacity of distributor *d* in period *t* for storing unprocessed product
δ_d^t	Surplus capacity of distributor *d* in period *t* for storing processed product
R_j	Lost sales of customer *j*
pr_d^t	Inventory level of distributor *d* in period *t*

2.3. Problem formulation

The paper considers a three echelons supply chain network with multiple suppliers, distributors, and customers. The formulation of the proposed multi-objective integer non-linear programming model is as follows:

$$maxz_1=\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{d'=1}^{D}\left(S_{dd'}^{Dt}-\frac{\sum_{t=1}^{T}\sum_{p=1}^{P}S_p^{Pt}}{TP}-C_d^{s}-\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}\left(C_{pdm}^{Tt}*U_{pdm}^{t}\right)}{\sum_{p=1}^{P}X_{pd}^{t}}-\frac{\sum_{d=1}^{D}\sum_{d'=1}^{D}\sum_{m=1}^{M}(C_{dd'm}^{Tt}*U_{dd'm}^{t})}{\sum_{p=1}^{P}X_{pd}^{t}}\right)*Y_{dd'}^{t}+\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{j=1}^{J}\left(S_{dj}^{Jt}-\frac{\sum_{t=1}^{T}\sum_{p=1}^{P}S_p^{Pt}}{TP}-C_d^{s}-\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}\left(C_{pdm}^{Tt}*U_{pdm}^{t}\right)}{\sum_{p=1}^{P}X_{pd}^{t}}-\frac{\sum_{d=1}^{D}\sum_{J=1}^{J}\sum_{m=1}^{M}(C_{djm}^{Tt}*U_{djm}^{t})}{\sum_{p=1}^{P}X_{pd}^{t}}+C_d^{Rt}-C_d^{F}\right)*Z_{dj}^{t} \quad (1)$$

$$minz_2=\sum_{t=1}^{T}\sum_{p=1}^{P}\left(1-\frac{S_p^{Pt}}{\sum_{p=1}^{P}S_p^{Pt}}\right)\left(\lambda_p^{Pt}-\sum_{d=1}^{D}X_{pd}^{t}\right) \quad (2)$$

$$minz_3=\sum_{t=1}^{T}\sum_{d=1}^{D}\left(W_d^{t}*\alpha_d^{t}\left(\frac{\sum_{p=1}^{P}S_p^{Pt}}{P}+C_d^{S}+\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}\left(C_{pdm}^{t}*U_{pdm}^{t}\right)}{\sum_{p=1}^{P}X_{pd}^{t}}\right)\right) \quad (3)$$

$$minz_4=\sum_{t=1}^{T}\sum_{d=1}^{D}\left(W_d^{t}*\alpha_d^{t}\left(\frac{\sum_{p=1}^{P}S_p^{Pt}}{P}+C_d^{S}+\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}\left(C_{pdm}^{t}*U_{pdm}^{t}\right)}{\sum_{p=1}^{P}X_{pd}^{t}}+C_d^{Rt}+C_d^{F}\right)\right) \quad (4)$$

S.t.

$$\left(\sum_{p=1}^{P}X_{pd}^{t}+\sum_{d'=1}^{D}Y_{d'd}^{t}\right)\le 0.8.\lambda_p^{Pt} \qquad \forall\, t,d \quad (5)$$

$$\left(\delta_d^{t}*C_d^{CSt}\right)+(\omega_d^{t}*C_d^{CFt})\le\ B_d^{t} \qquad \forall\, t,d \quad (6)$$

$$\sum_{p=1}^{P}X_{pd}^{t}+W_d^{t-1}+\sum_{d'=1}^{D}Y_{d'd}^{t}=W_d^{t}+\sum_{d'=1}^{D}Y_{dd'}^{t} \qquad \forall d,t\ge 2 \quad (7)$$

$$\sum_{p=1}^{P}X_{pd}^{t}+\sum_{d'=1}^{D}Y_{d'd}^{t}=W_d^{t}+Pr_d^{t}+\sum_{d'=1}^{D}Y_{dd'}^{t} \qquad \forall d,t=1 \quad (8)$$

$$Y_{dd'}^{t}=0 \qquad \forall\, t,d=d' \quad (9)$$

$$W_d^{t}\le\ \sum_{p=1}^{P}X_{pd}^{t} \qquad \forall\ t=1,d \quad (10)$$

$$\sum_{d'=1}^{D}Y_{dd'}^{t}\le W_d^{t} \qquad \forall\, t,d \quad (11)$$

$$K_d^{t-1}+Pr_d^{t}=K_d^{t}+\sum_{j=1}^{J}Z_{dj}^{t} \qquad \forall\ t\ge 2,d \quad (12)$$

$$Pr_d^{1}=K_d^{1}+\sum_{j=1}^{J}Z_{dj}^{1} \qquad \forall d \quad (13)$$

$$\sum_{d=1}^{D}X_{pd}^{t}\le\lambda_p^{Pt} \qquad \forall\, t,p \quad (14)$$

$$K_d^{t}\le\lambda_d^{Rt} \qquad \forall\, t,d \quad (15)$$

$$\sum_{d=1}^{D}Z_{dj}^{t}\le\lambda_j^{J^-t} \qquad \forall\, j,t \quad (16)$$

$$\sum_{d=1}^{D}Z_{dj}^{t}\ge\lambda_j^{J^-t} \qquad \forall\, j,t \quad (17)$$

$$\sum_{t=1}^{T}\sum_{d=1}^{D} Z_{dj}^{t} = \sum_{t=1}^{T} \lambda_j^{Jt} + R_j \qquad \forall\, j \tag{18}$$

$$K_d^t \le \lambda_d^F + \delta_d^t \qquad \forall\, t, d \tag{19}$$

$$W_d^t \le \lambda_d^S + \omega_d^t \qquad \forall\, t, d \tag{20}$$

$$\left(\frac{\sum_{t=1}^{T}\sum_{p=1}^{P} S_p^{Pt}}{TP} + C_d^S + \frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}(C_{pdm}^{Tt} * U_{pdm}^{t})}{\sum_{p=1}^{P} X_{pd}^{t}} + C_d^{Rt} + C_d^F\right) * K_d^t \le B_d^t \qquad \forall\, t, d \tag{21}$$

$$X_{pd}^t \le V_m * U_{pdm}^t \qquad \forall\, t, p, d, m \tag{22}$$

$$Y_{dd'}^t \le V_m * U_{dd'm}^t \qquad \forall\, t, d, d', m \tag{23}$$

$$Z_{dj}^t \le V_m * U_{djm}^t \qquad \forall\, t, j, d, m \tag{24}$$

$$K_d^t = 0 \qquad \forall\, d, t = 1 \tag{25}$$

$$W_d^t = 0 \qquad \forall\, d, t = 1 \tag{26}$$

$$X_{pd}^t, Y_{dd'}^t, Z_{dj}^t, K_d^t, W_d^t, U_{pdm}^t, U_{dd'm}^t, U_{djm}^t, \omega_d^t, \delta_d^t, R_j, pr \in I \quad \forall t, d, m, j, p, d' \tag{27}$$

The first objective function (1) aims to maximize the total profit of the network by optimizing the profit of the distributors. Distributor cans sales unprocessed products to other distributors and processed products to customers. Hence, the first term of the equation (1) calculates the multiplication of the amount of the unprocessed product sold to other distributors with the unit profit of an unprocessed product. Also, the second term obtains the multiplication of the amount of the processed product sold to customers with the unit profit of a processed product. To this end, the unit profit of the unprocessed or processed product were obtained from the difference in sale price and average cost including average purchase price from producer or other distributors, warehouse costs, the transportation cost between nodes of the network. The total amount of the product that was not bought from producer through harvest period was minimized by objective function (2). Objective functions (3) and (4) minimizing the perished cost of the stored processed and unprocessed product respectively. It should be noted that the perished rate of the processed and unprocessed citrus are different in warehouse.

Constraint (5) guarantees that total useless processing capacity of each distributor were not less than eighty percent of total processing capacity. It means that if a distributor facility is setup at least eighty percent of its processing capacity should be activated in each period. Constraint (6) restricts the total budget for renting warehouses by distributors at each period. The distributors can store surplus processed or unprocessed product with cost C_d^{CSt} and C_d^{CFt} for an unprocessed and processed product respectively. Constraints (7), (8), (9), (10) and (11) are the balance constraints between producers and distributors and distributors and distributors. They are guarantee that the processing and sale of each distributor cannot be more than of total amount of purchased unprocessed product. Constraints (12) and (13) balance the flow of the processed product between distributors and customers through planning horizon. The capacity of the suppliers and distributors are restricted with constraints (14) and (15). Constraints (16) and (17) ensure that the total amount of product delivery to the customer would be between the lower and upper limit of the customer demand. A part of customers' demand might be lost and equality (18) considers the total lost sale of customers. Constraint (19) and (20) ensure that total stored unprocessed and processed product should be less than warehouse capacity of the distributors at each period. However, the distributors could store more products by paying surplus cost (see constraint 6). Constraint (21) ensure that the total cost of the stored products be less than total budget of the distributors. Constraint (22), (23) and (24) restrict the number of used transportation equipment according to their capacity. Constraints (25), (26) guarantee that the inventory of the unprocessed and processed product be zero at the end of the planning horizon. Finally, constraint (27) enforces the integer and non-negativity restrictions on corresponding decision variables.

2.4. Linearization

To solve the proposed non-linear integer programming model, the linearization process has been utilized. While the linear models can be reached to global solution, often the local solution was obtained from nonlinear models. In the proposed model, the formulas (1), (3), (4), and (21) are nonlinear. To convert the proposed non-linear model into a linear one, the presented method by (Mahdavi et al., 2012) has been used. New variables are added to model and linear equations are generated as follows:

$$\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}U_{pdm}^{t}}{\sum_{p=1}^{P}X_{pd}^{t}}\times Y_{dd'}^{t}=AB_{d}^{t} \tag{28}$$

$$\frac{\sum_{d=1}^{D}\sum_{d'=1}^{D}\sum_{m=1}^{M}U_{dd'm}^{t}}{\sum_{p=1}^{P}X_{pd}^{t}}\times Y_{dd'}^{t}=AC_{d}^{t} \tag{29}$$

$$\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}U_{pdm}^{t}}{\sum_{p=1}^{P}X_{pd}^{t}}\times Z_{dj}^{t}=AD_{d}^{t} \tag{30}$$

$$\frac{\sum_{d=1}^{D}\sum_{j=1}^{J}\sum_{m=1}^{M}U_{pdm}^{t}}{\sum_{p=1}^{P}X_{pd}^{t}}\times Z_{dj}^{t}=AE_{d}^{t} \tag{31}$$

$$\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}U_{pdm}^{t}}{\sum_{p=1}^{P}X_{pd}^{t}}\times W_{d}^{t}=AF_{d}^{t} \tag{32}$$

$$\begin{aligned}\max z1=&\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{d'=1}^{D}\left(((S_{dd'}^{Dt}-\frac{\sum_{t=1}^{T}\sum_{p=1}^{P}S_{p}^{PT}}{T.P}-C_{d}^{S}-\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}C_{pdm}^{Tt}}{\sum_{p=1}^{P}X_{pd}^{t}}-\frac{\sum_{d=1}^{D}\sum_{d'=1}^{D}\sum_{m=1}^{M}C_{pd'm}^{Tt}}{\sum_{p=1}^{P}X_{pd}^{t}})\times Y_{dd'}^{t})-AB_{d}^{t}-AC_{d}^{t}\right)+\\&\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{j=1}^{J}\left(((S_{dj}^{Jt}-\frac{\sum_{t=1}^{T}\sum_{p=1}^{P}S_{p}^{PT}}{T.P}-C_{d}^{S}-\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}C_{pdm}^{Tt}}{\sum_{p=1}^{P}X_{pd}^{t}}-\frac{\sum_{d=1}^{D}\sum_{j=1}^{j}\sum_{m=1}^{M}C_{djm}^{Tt}}{\sum_{p=1}^{P}X_{pd}^{t}})\times Z_{dj}^{t})-AD_{d}^{t}-AE_{d}^{t}\right)\end{aligned} \tag{33}$$

$$\min z3=\sum_{t=1}^{T}\sum_{d=1}^{D}\left(W_{d}^{t}\times\alpha'_{d}\left(\frac{\sum_{p=1}^{p}S_{p}^{PT}}{P}+C_{d}^{S}+\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}C_{pdm}^{Tt}}{\sum_{p=1}^{P}X_{pd}^{t}}\right)+\alpha'_{d}\left(AF_{d}^{t}\right)\right) \tag{34}$$

$$\min z4=\sum_{t=1}^{T}\sum_{d=1}^{D}\left(W_{d}^{t}\times\alpha_{d}\left(\frac{\sum_{p=1}^{p}S_{p}^{PT}}{P}+C_{d}^{S}+\frac{\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}C_{pdm}^{Tt}}{\sum_{p=1}^{P}X_{pd}^{t}}+C_{d}^{Rt}+C_{d}^{F}\right)+\alpha_{d}\left(AF_{d}^{t}\right)\right) \tag{35}$$

$$\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}U_{pdm}^{t}+\sum_{p=1}^{P}X_{pd}^{t}+W_{d}^{t}\geq AF_{d}^{t} \tag{36}$$

$$\sum_{p=1}^{P}\sum_{d=1}^{D}\sum_{m=1}^{M}U_{pdm}^{t}+\sum_{p=1}^{P}X_{pd}^{t}+W_{d}^{t}\leq AF_{d}^{t} \tag{37}$$

3. Solution Methodology

To solve the proposed non-linear integer programming model, the linearization process has followed. While the linear models can be reached to global solution, often the local solution was obtained from nonlinear models. In the proposed model, the formulas (1), (3), (4), and (21) are nonlinear. To convert the proposed non-linear model into a linear one, the presented method by (Mahdavi et al., 2012) has been used. New variables are added to model and converted equations are generated as follows:

3.1. NSGA-II

NSGA-II was introduced by Deb et al. (2002) by developing NSGA algorithm. Advantage of NSGA-II algorithm is that it is more efficient computationally (Afshar-Nadjafi, & Razmi-Farooji, 2016). Moreover, this method uses elitism and crowded comparison operator that can survive variance without using parameter addition Deb et al. (2002). Unlike numerical processing methods, the NSGA-II could solve the multi-objective models with a single run. Since the optimality of the whole objective functions does not exist in with a single solution, the NSGA-II provides a set of the Pareto solutions by taking into account the diversification features.

Here, two phases have been followed for applying the NSGA-II algorithm. The first phase considers the quality of solutions by determining the ranks of the solutions. To this end two values are calculated including the number of times that a solution will be overcome and the set of solutions recessive by the current answer. To determine these two values, all the solutions should be compared with each other. Solutions with failure count zero are unbeaten solutions and add to the Pareto front 1 (F1). The solutions of the second front are determined by subtracting the value 1 from the number of the whole recessive answers. In the second phase we used the distance congestion measure, which is indicates the distance between all solutions of a same level. The greater the selective points crowding distance is, the more they help to the variety. Comparing two different answers, we are faced with two modes consisting of a) between two solutions of different ranks, the solution with lesser rank excels, b) If two solutions are in same front, the solution with the more crowding distance is preferred.

If P_t be the current generation population and Q_t be number of children, which have been created using crossover and mutation operators, to create the next generation first the answers of Q_t and P_t are merged and then sorting is done over this merged population using the ranking function and then crowding distance. Finally, the number of the first arranged population size transfer to the next generation directly and the rest Solutions will be deleted. In fact, the algorithm balances between quality and order using the high importance it gives to the solution ranks and the less to crowding distance (Trisna, et al., 2016). The following steps are considered to solve the proposed model with NSGA-II algorithm:

1. Initialize population as usual.
2. Create a random parent population P0 of size N. Set t =0.
3. Use genetic operator (crossover and mutation) toward P_0 to create offspring population Q_0 of size N
4. If the stopping criterion is satisfied, then stop and return to P_t .
5. Set $R_t = P_t \cup Q_t$.
6. Rank population and identify the non-dominated fronts $F_1, F_2, ..., F_R$ in R_t applying the fast non-dominated sorting algorithm, The first front (F_1) is a non-dominant set for current population and the second front (F_2) is dominated by the individuals in the first front only and goes so on for the next front.
7. For each objective function k, sort the solutions in F_j in the ascending order. Let l =| F_j | and x(i,j) represent the *i*th solution in the sorted list with respect to the objective function k. Assign cd_k ($X_{[1,k]}$)= ∞ and cd_k ($X_{[1,k]}$)= ∞ and for i = 2,…, l-1 assign.

$$cd_k(X_{[1,k]}) = \frac{Z_K(X_{[i+1,k]}) - Z_K(X_{[i-1,k]})}{Z_K^{max} - Z_K^{min}} \quad (38)$$

Initially, we generate a number of random solutions and select the best one as the current solution. We utilized five neighborhood search structures as presented in Figure 1.

3.2. MOPSO

Particle swarm optimization is an evolutionary computation techniques introduced by Eberchart and Kennedy (1995) as social behavior. PSO algorithm starts searching a particle population and keeps surviving for all generations until searching stop criteria is met. Each particle has some memory which helps to track the best position it has acquired so far and the best position any other particle acquired so far within the neighborhood. The particle will then modify its direction based on components towards its own best position and towards the overall best position (Fattahi, & Samouei, 2016). Each individual (particle) represents a solution in an n-dimensional space. Besides, each particle also has knowledge of its prior and the best experience and knows the global best solution found by the entire swarm. Each particle updates its way using the equations as follows:

$$V_{ij} = W * V_{ij} + c_1 * r_1(P_{ij} - X_{ij}) + c_2 * r_2(P_{gj} - X_{ij}) , \quad X_{ij} = X_{ij} + V_{ij} \quad (39)$$

Where w is the inertia factor affecting the local and global capabilities of the algorithm, vij is the velocity of the particle i in the jth dimension, c_1 and c_2 are weights affecting the cognitive and social factors, respectively. r_1 and r_2 are uniform random variables between 0 and 1. P_{ij} is the best value found by particle i (the best of p) and P_{gj} is the global best found by the entire swarm (the best of g).

To solve the proposed multi-objective model of citrus supply chain network, we modified the PSO algorithm with LP-metric method (Mirzapour Al-E-Hashem, Malekly, & Aryanezhad, 2011). The LP-metric method requires the optimum value of each objective (f_j). It intends to minimize the total weighted deviations from the ideal value of each objective (f_j). To evaluate the fitness of a solution, this method uses the following formula:

$$LP=\left[\sum_{\forall j}\lambda_j\left(\frac{f_j^*-f_j}{f_j^*}\right)^p\right]^{\frac{1}{p}} \tag{40}$$

Where f_j and λ_j are the value and weight of jth objective, respectively. P is a control parameter that takes an integer value equal to or greater than 1. If we have $p = \infty$, the problem becomes the minimization of the maximal deviation as follows:

$$LP = Min\left\{\max\left\{\lambda_1\left(\frac{f_1^*-f_1}{f_1^*}\right),\lambda_2\left(\frac{f_2^*-f_2}{f_2^*}\right),\dots,\lambda_k\left(\frac{f_k^*-f_k}{f_k^*}\right)\right\}\right\} \tag{41}$$

As a result, the problem can be stated as follows:

$$\begin{aligned}&Min\,z\\&s.t.\\&z\geq\lambda_j\left(\frac{f_j^*-f_j}{f_j^*}\right),\ \forall j\end{aligned} \tag{42}$$

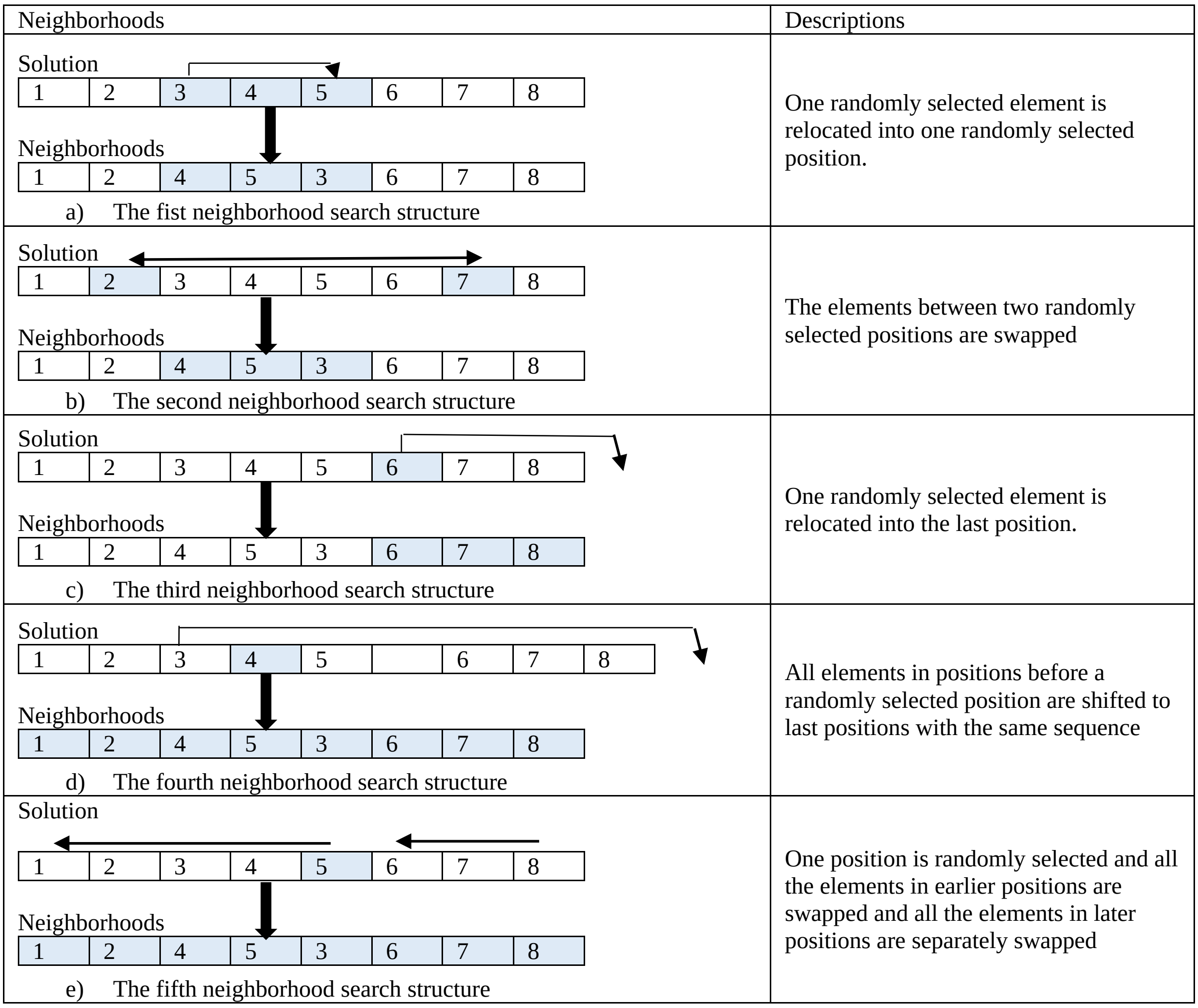

Fig. 1. The examples of the five neighborhood search structure.

3.3. MOICA

ICA Algorithm was developed by Atashpas-Gargari and Lucas (2007) as a population based evolutionary algorithm. The inspiration of the algorithm comes from the socio-political process of imperialistic competition in the real world. The MOICA establishes two concepts of non-dominance and crowding distance. In this algorithm, numbers of random solutions, named countries, are firstly generated to form the initial population. After producing the countries, a non-dominance technique and a crowding distance measure is utilized to rank and select the best countries in the population. In each generation, the best solutions are selected as the imperialists and the remaining countries are treated as the colonies Naderi, (2013).

All the colonies of initial countries are divided among the mentioned imperialists based on their power. The imperialist and its countries make an empire. The power of the imperialist country is the key factor to determine total power of an empire. Imperialists persuade their colonies to move toward themselves. The information of the colonies is shared by crossover operation and the imperialism affects is of colonies by mutation operation. This process continues until the power of the stronger empire is increased and the power of the weaker ones is reduced. As the algorithm goes on, the strongest imperialists take up the colonies of less powerful imperialists and the weak empires will be deleted. The algorithms stop as soon just one emperor remains (Alaghebandha, Pasandideh & Hajipour, 2012).

4. Experimental Results

To evaluate the performance of the NSGA-II, MOICA, and MOPSO, we consider 14 small and large size test problems as reported in the Appendix I. For each test problem 7 instances was generated randomly and summarized in the Appendix II. The algorithms are coded in MATLAB V. 12.10.0.499 and run on a Pentium IV 2.5 GHz processor with 6 GB memory. It should be noted that only the small size test problems could be solved by Branch and Bound Algorithm (computational time about 80–550 second in LINGO 8.0 software) due to the computational complexity of the proposed model. For the large size test problems, we cannot obtain any feasible solutions with Branch and Bound Algorithm after 1200s. The average computational times of meta-heuristic algorithms were obtained about 54.68 s and 262.51s for small and large test problems respectively and shown in figure 2.

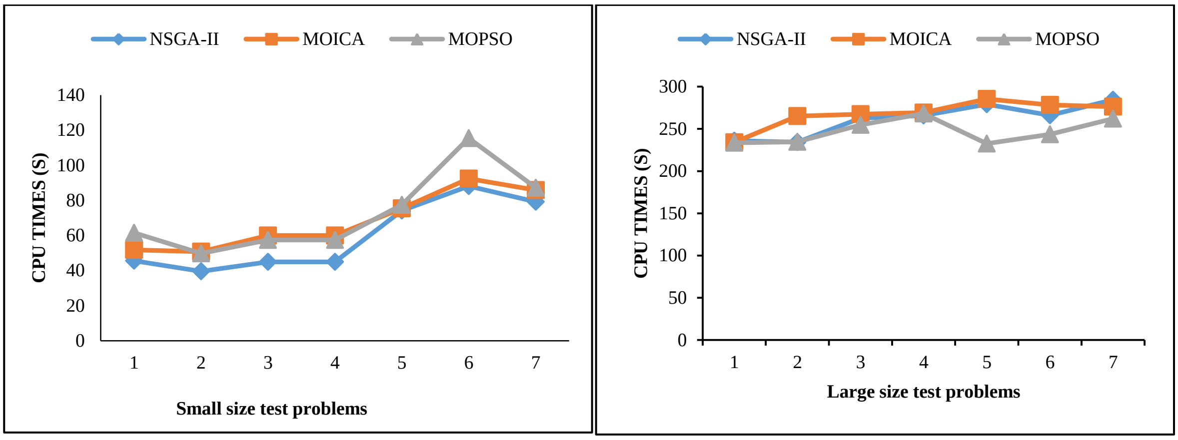

Fig. 2. The computational time of small and large size test problems for each algorithm

We utilize relative percentage deviation (RPD) criteria to compare the performance of MOICA, NSGA-II, and MOPSO algorithms. To calculate RPD for minimization and maximization objective function the following equations were utilized:

$$RPD = \frac{Alg_{sol} - min_{sol}}{min_{sol}} \times 100$$
$$RPD = \frac{max_{sol} - Alg_{sol}}{max_{sol}} \times 100 \qquad (43)$$

where *Alg*, *Max*, and *Min* were the solutions of the algorithm, and the optimum solution if the problem is solved in single objective manner with maximization and minimization. Tables 1 report the average RPD of algorithms. The results of the Branch and Bound (B&B) algorithm were reported for small size due to the computational time more than 1200s. Figure 3 shows the average RPD of objective functions obtained for each test problem. Indeed, the RPD of B&B algorithm for small size test problems is the best one. Among the meta-heuristic algorithms, MOPSO obtained the better RPD value rather than MOICA and NSGA-II in both small and large size test problems as shown in figure 3 and summarized in table 1.

Table 1
The average RPD of NSGA-II, MOICA, and MOPSO on different problems

Test problems	NSGA-II	MOPSO	MOICA	B&B
SS1	0.968	0.544	0.821	0.342
SS2	1.355	0.877	1.042	0.632
SS3	2.972	1.655	2.832	1.295
SS4	3.763	1.876	2.733	1.693
SS5	4.971	2.987	3.252	2.576
SS6	5.506	3.286	4.261	3.096
SS7	5.931	3.901	4.988	3.344
LS1	5.786	4.122	4.984	-
LS2	6.445	5.347	5.972	-
LS3	7.866	6.191	6.899	-
LS4	8.788	6.983	7.981	-
LS5	9.556	7.173	8.456	-
LS6	9.788	7.934	8.934	-
LS7	10.988	8.283	10.145	-

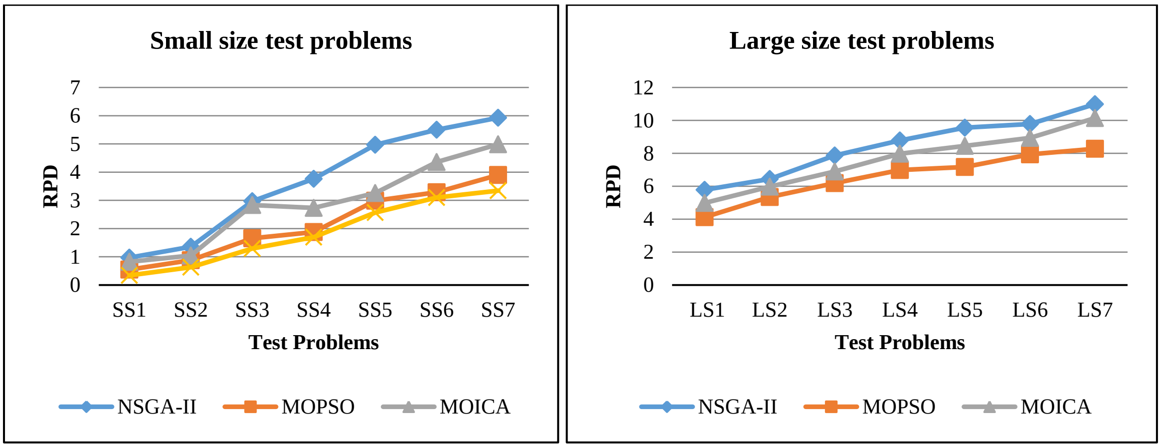

Fig. 3. The integrated RPD of NSGA-II, MOICA, and MOPSO for different problem sizes

4.1. Results validation

The performance of the meta-heuristic algorithms depends on the parameters that were used in the components of them. So the validity of the solutions must be evaluated to ensure the real Pareto front. To this end, three small size test problems were generated and the real Pareto front space was found. The real Pareto front is compared with the result of the applied algorithms. The detailed results of three generated examples were summarized in table 2. As reported in table 2, the NSGA-II found more non-dominated solutions in the first and third examples (i.e. 15 and 14). We can found 39 non-dominated solutions by MOPSO algorithm for the second test problem. The value of objective functions for the first example that was obtained by NSGA-II (15 non-dominated solution) is reported in Appendix III. Distribution of points in the solution space is shown in figure 4. According to the obtained results, we can conclude that the accuracy and the ability of the three algorithms in achieving Pareto front solutions are acceptable.

Table 2
Characteristics and results for examples produced for validation algorithms

Test Problem	The total number of points possible solution space	The total number of points Non-dominated	The total solution time (s)	The number of Non-dominated solution			The running time of three algorithms (s)
				MOPSO	NSGA-II	MOICA	
1	4850	17	274	12	15	13	5
2	37487	38	24560	39	36	31	35
3	196534	13	23730	10	14	12	40

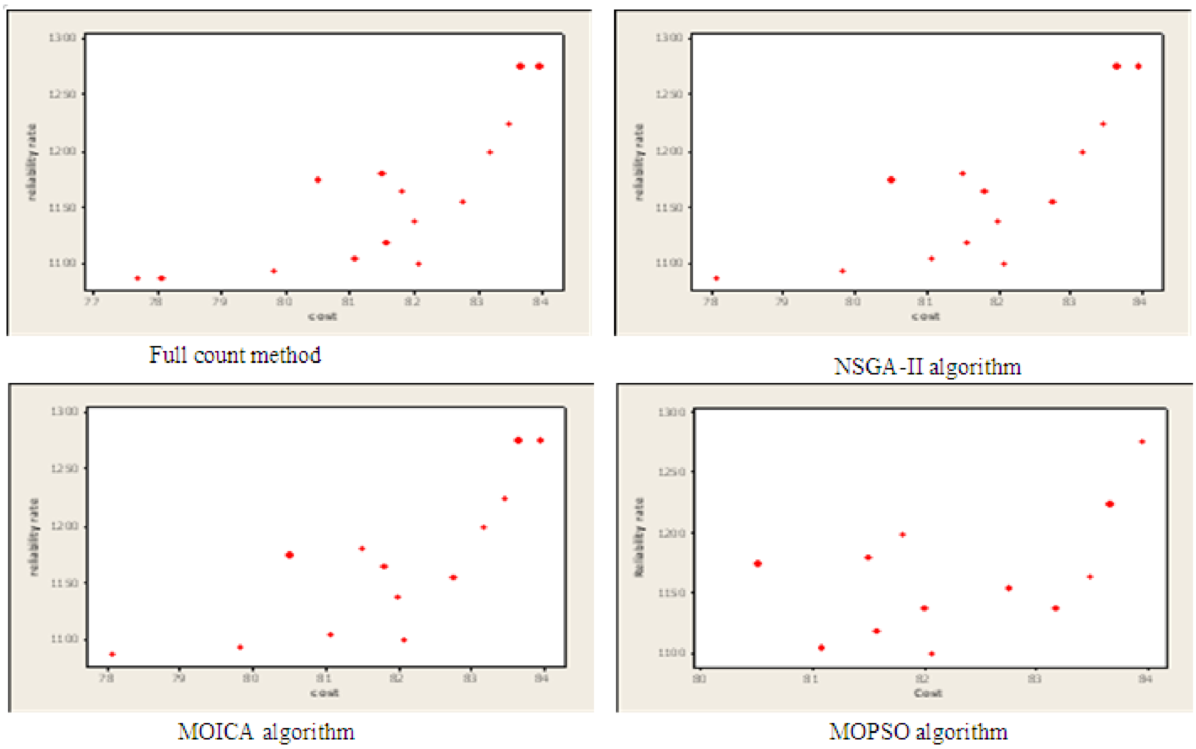

Fig. 4. Pareto points dispersion in the solution space

4.2. Comparison of Algorithms

To compare the performance of the multi-objective meta-heuristic algorithms, several indicators were proposed in the literature (Afshar-Nadjafi, & Razmi-Farooji, 2014 and Sarrafha, Kazemi, & Alinezhad, 2014). Generally, indicators are evaluating two main features of multi-objective programming consisting of the convergence and quality and distribution and expansion of solutions in the solution space. We used four indicators to compare the utilized algorithms including Number of Pareto Solution (NOS), Maximum Spread (MS), Mean Ideal Distance (MID), and Diversification Metric (DM). To this end the identified indicators were calculated for MOPSO, MOICA, and NSGA-II algorithms in small and large size test problems.

- *Small size test problems:*

Four small size test problems were generated for comparing the algorithms. The results show that MOPSO algorithm is more successful than NSGA-II and MOICA algorithms for finding the number of Pareto points in all problems. In relation to the average distance from the ideal answer a definite statement cannot be given about the superiority of any ones over the other algorithms. For diversification metric and maximum spread criteria which have an indication over distribution of solutions superiority MOPSO algorithm is better than MOICA and NSGA-II algorithms.

- *Large size test problems:*

Three large size test problems were generated to compare the algorithms. The values of the indicators were reported in table 4. The MOPSO has shown better results according to the number of Non-dominant solutions. By increasing the size of the problems, NSGA-II algorithm finds more Pareto points in compare to MOICA algorithm. However, MOICA algorithm was converged and Non-dominant solutions were obtained rapidly. It should be noted that we cannot find new non-dominated solutions by increasing the number of the iterations in MOICA algorithm. As reported in table 4, we found out the NSGA-II is a capable algorithm for solving the proposed model in large size test problem. Although the computational time of the NSGA-II was more than two other ones. Based on the computational time, the MOICA showed better performance rather than NSGA-II and MOPSO for large size test problem. We can claim that the MOPSO algorithm has acceptable running time as well as four indicators in large size test problems.

Table 3
Comparison of the algorithms for small size test problems

Test problem	algorithm	diversification metric	Mean ideal distance	Maximum spread	Number of Pareto solution	Run time (s)
1	NSGA-II	12/32	51/43	114/43	30	10
	MOICA	12/5	47/76	115/22	31	10
	MOPSO	11/98	39/74	105/32	29	10
2	NSGA-II	10/06	34/27	87/63	24	10
	MOICA	10/24	40/07	90/54	25	10
	MOPSO	9/11	31/47	85/35	23	10
3	NSGA-II	8/94	37/07	65/39	19	30
	MOICA	9/02	33/79	67/31	34	30
	MOPSO	8/15	29/3	63/01	23	30
4	NSGA-II	22/96	202/7	396/14	132	150
	MOICA	24/01	214/42	440/19	126	150
	MOPSO	21/45	198/65	432/85	116	150

Table 4
Comparison of the algorithms for large scale test problems

Test Problems	Algorithm	diversification metric	Mean ideal distance	Maximum spread	Number of Pareto solution	Run time (s)
1	NSGA-II	20/05	115/14	354/5	52	600
	MOICA	23/4	158/09	566/84	49	600
	MOPSO	24/64	165/16	576/6	47	600
2	NSGA-II	22/67	183/76	546/23	104	900
	MOICA	27/16	271/32	753/56	53	900
	MOPSO	25/29	289/45	776/43	48	900
3	NSGA-II	22/45	214/08	494/14	114	1000
	MOICA	26/07	249/95	654/5	43	1000
	MOPSO	28/79	256/76	697/3	39	1000

5. Conclusions and Future Research Directions

This paper developed a three-echelon citrus supply chain network model as a multi-objective single-product multi-period location-allocation problem. The proposed multi-objective integer non-linear programming model aims to maximize the profit of the networks while minimize the total costs. Three meta-heuristic algorithms including MOPSO, MOICA, and NSGA-II were coded to solve the citrus SCND problem. To evaluate the performance of the algorithms several test problems have been generated randomly with different sizes. First, computational time and RPD criteria were used to evaluate the performance of algorithms. Then four multi-objective indicators i.e. number of Pareto solution, maximum spread, mean ideal distance, and diversification metric have been utilized for comparing the algorithms in small and large size test problems. The following results were found from solving the proposed citrus SCND problem by three multi-objective meta-heuristics algorithms:

- The computational time of the NSGA-II algorithm would be increased significantly by increasing the size of the test problems;
- The computational time of both MOPSO and MOICA were same relatively while in large size test problems the MOPSO show better performance.;
- According to RPD criteria, the MOPSO was better than two other algorithms for solving the proposed citrus SCND problem. In addition, in small size test problems, the RPD of the MOPSO is close to the results of the B&B algorithm;
- Based on the Pareto point dispersion criteria, the NSGA-II algorithm is more similar to full count method rather than MOPSO and MOICA algorithms;
- In the large size test problems, the MOICA was converged rapidly while the number of its Parreto solution was less than NSGA-II algorithm;
- Based on the comparison indicators and performance criteria we can claim that the MOPSO has acceptable results for solving the proposed citrus SCND problem.

As directions for future researches, the model presented in this paper could be implemented in a real case study to evaluate the applicability of the proposed models and solution methods. In addition, other solution approaches could be utilized to solve the developed problem in different sizes.

Appendix I. Test Problems size

The sets of test problems

Test Problems	Sets				
	Producer	Distributor	Customer	Period	Transportation equipment
Small					
SS1	2	2	2	1	1
SS2	2	2	2	2	1
SS3	1	2	2	2	2
SS4	1	2	3	3	2
SS5	2	2	3	2	2
SS6	2	2	3	3	3
SS7	2	2	3	2	1
Large					
LS1	8	10	15	4	5
LS2	15	18	16	4	5
LS3	17	18	16	4	5
LS4	19	20	17	4	6
LS5	20	22	20	5	7
LS6	20	23	20	5	6
LS7	25	24	23	4	5

Appendix II. Test Problems parameters' distributions

Parameters' distribution

Parameters	Problem ranges	Parameters	Problem ranges
S_p^{Pt}	[150،100]	$C_{dd'm}^{Tt}$	[2,10]
$S_{dd'}^{Dt}$	[120،100]	C_{pjm}^{Tt}	[2,10]
S_{dj}^{Jt}	[450, 500]	C_d^{Rt}	[5,15]
λ_p^{Pt}	[200,250]	C_d^S	[5,10]
λ_d^S	[100,200]	C_d^F	[5,10]
λ_d^F	[150،100]	C_d^{CSt}	[5,10]
λ_d^R	[200,250]	C_d^{CFt}	[10,20]
λ_j^{Jt}	[150،100]	V_m	[1000,2000]
$\lambda_j^{J^+t}$	[150،100]	α'_d	[0/00015,0/00085]
$\lambda_j^{J^-t}$	[150،100]	α_d	[0/00025,0/00075]
C_{pdm}^{Tt}	[5,10]		

Appendix III. The values of objective functions related to the non-dominant first example solved by NSGA-II

The point of non-dominated	Costs average	Distributor profit
1	83.96	1275
2	83.96	1224
3	83.48	1163
4	81.8	1198
5	83.18	1137
6	82	1137
7	81.5	1179
8	82.76	1154
9	81.57	1118
10	81.07	1104
11	80.51	1174
12	82.07	1099
13	79.82	1093
14	78.07	1087
15	77.7	1087

Reference

Afshar-Nadjafi, B. & Razmi-Farooji, A. (2014). A Comparison of NSGA II and MOSA for Solving Multi-depots Time-dependent Vehicle Routing Problem with Heterogeneous Fleet. *Journal of Optimization in Industrial Engineering*, 7(16): p. 65-73.

Afshar-Nadjafi, B. & Razmi-Farooji, A. (2016). Using NSGA II and MOSA for solving multi-depots time-dependent vehicle routing problem with heterogeneous fleet. *Journal of Optimization in Industrial Engineering*,

Ahumada, O. & Villalobos, J.R. (2011).Operational model for planning the harvest and distribution of perishable agricultural products. *International Journal of Production Economics*, 133(2): p. 677-687.

Alaghebandha, M., Pasandideh, S.H.R. % Hajipour, V. (2012). A Continuous Review inventory Control Model within Batch Arrival Queuing Framework: A Parameter-Tuned Imperialist Competitive Algorithm. *Journal of Optimization in Industrial Engineering*, 5(11): p. 11-24.

Atashpaz-Gargari, E. & Lucas, C. (2007). Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. in Evolutionary computation, 2007. CEC 2007. IEEE Congress on. IEEE.

Bortolini, M., et al., (2016). Fresh food sustainable distribution: cost, delivery time and carbon footprint three-objective optimization. *Journal of Food Engineering*, 174: p. 56-67.

Bozorgi-Amiri, A., et al., (2012). A modified particle swarm optimization for disaster relief logistics under uncertain environment. *The International Journal of Advanced Manufacturing Technology*, 60(1): p. 357-371.

Christopher, M. (2016). Logistics & supply chain management: Pearson UK.

Deb, K., et al., (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE transactions on evolutionary computation, 6(2): p. 182-197.

Derwik, P. & Hellström, D. (2017). Competence in supply chain management: A systematic review. Supply Chain Management: *An International Journal*, 22(2).

Eberchart, R. & Kennedy, J. (1995). Particle swarm optimization. in IEEE International Conference on Neural Networks, Perth, Australia.

Eberhart, R. & Kennedy, J. (1995). A new optimizer using particle swarm theory. in Micro Machine and Human Science, MHS'95., Proceedings of the Sixth International Symposium on. 1995: IEEE.

Fattahi, P. & Samouei, P. (2016). A Multi-Objective Particle Swarm Optimization for Mixed-Model Assembly Line Balancing with Different Skilled Workers. *Journal of Optimization in Industrial Engineering*, 9(20): p. 9-18.

Ghodratnama, A., Jolai, F. & Tavakkoli-Moghaddam, R. (2015) Solving a new multi-objective multi-route flexible flow line problem by multi-objective particle swarm optimization and NSGA-II. *Journal of Manufacturing Systems*, 36: p. 189-202.

González-Araya, M.C., Soto-Silva, W.E. & Espejo, L.G.A. (2015). Harvest Planning in Apple Orchards Using an Optimization Model, in Handbook of Operations Research in Agriculture and the Agri-Food Industry, Springer. p. 79-105.

Govindan, K., (2017). Sustainable consumption and production in the food supply chain: A conceptual framework. *International Journal of Production Economics*.

Govindan, K., et al., (2014). Two-echelon multiple-vehicle location–routing problem with time windows for optimization of sustainable supply chain network of perishable food. *International Journal of Production Economics*, 152: p. 9-28.

Hafezalkotob, A., K. (2016).Khalili-Damghani, and S.S. Ghashami, A Three-Echelon Multi-Objective Multi-Period Multi-Product Supply Chain Network Design Problem: A Goal Programming Approach. *Journal of Optimization in Industrial Engineering*, 10(21): p. 67-78.

Harris, I., Mumford, C.L. & Naim, M.M. (2014). A hybrid multi-objective approach to capacitated facility location with flexible store allocation for green logistics modeling. Transportation Research Part E: Logistics and Transportation Review, 66: p. 1-22.

Ju, M., M. Osako, and S. Harashina, (2017). Food loss rate in food supply chain using material flow analysis. *Waste Management*, 61: p. 443-454.

Khalili-Damghani, K., A.-R. Abtahi, & Tavana, M. (2013). A new multi-objective particle swarm optimization method for solving reliability redundancy allocation problems. Reliability Engineering & System Safety, 111: p. 58-75.

Latha Shankar, B., et al., (2013). Location and allocation decisions for multi-echelon supply chain network – A multi-objective evolutionary approach. *Expert Systems with Applications*, 40(2): p. 551-562.

Litchy, A.J. & Nehrir, M.H. (2014). Real-time energy management of an islanded microgrid using multi-objective Particle Swarm Optimization. in 2014 IEEE PES General Meeting | Conference & Exposition.

Lowe, T.J. & Preckel, P.V. (2004). Decision technologies for agribusiness problems: A brief review of selected literature and a call for research. *Manufacturing & Service Operations Management*, 6(3): p. 201-208.

Mahdavi, I., et al., (2012). A new mathematical model for integrating all incidence matrices in multi-dimensional cellular manufacturing system. *Journal of Manufacturing Systems*, 31(2): p. 214-223.

Manders, J.H.M., Caniëls, M.C.J. & Ghijsen, P.W.T. (2016). Exploring supply chain flexibility in a FMCG food supply chain. *Journal of Purchasing and Supply Management,* 22(3): p. 181-195.

Melo, M.T., Nickel, S. & Saldanha-Da-Gama, F. (2009). Facility location and supply chain management–A review. *European journal of operational research*, 196(2): p. 401-412.

Mirzapour Al-E-Hashem, S., H. Malekly, & Aryanezhad, M. (2011). A multi-objective robust optimization model for multi-product multi-site aggregate production planning in a supply chain under uncertainty. *International Journal of Production Economics,* 134(1): p. 28-42.

Mousavi, S.M., et al., (2017). A modified particle swarm optimization for solving the integrated location and inventory control problems in a two-echelon supply chain network. *Journal of Intelligent Manufacturing*, 28(1): p. 191-206.

Nadal-Roig, E. & L.M. (2015). Plà-Aragonés, Optimal Transport Planning for the Supply to a Fruit Logistic Centre, in Handbook of Operations Research in Agriculture and the Agri-Food Industry., Springer. p. 163-177.

Naderi, B., (2013).The project portfolio selection and scheduling problem: mathematical model and algorithms. *Journal of Optimization in Industrial Engineering,* 6(13): p. 65-72.

Negi, S. & Anand, N. (2015). Issues and challenges in the supply chain of fruits & vegetables sector in India: A Review. *International Journal of Managing Value and Supply Chains*, 6(2): p. 47-62.

Pishvaee, M.S. (2017). Honey global supply chain network design using fuzzy optimization approach. *Journal of Industrial and Systems Engineering,* 10(3): p. 0-0.

Saeedi Mehrabad, M., Aazami, A. & Goli, A. (2017). A location-allocation model in the multi-level supply chain with multi-objective evolutionary approach. *Journal of Industrial and Systems Engineering*, 10(3): p. 0-0.

Sarrafha, K., Kazemi, A. & Alinezhad, A. (2014). A multi-objective evolutionary approach for integrated production-distribution planning problem in a supply chain network. *Journal of Optimization in Industrial Engineering,* 7(14): p. 89-102.

Soto-Silva, W.E., et al., (2016). Operational research models applied to the fresh fruit supply chain. *European Journal of Operational Research,* 251(2): p. 345-355.

Torabi, S.A., et al., (2013). A particle swarm optimization for a fuzzy multi-objective unrelated parallel machines scheduling problem. *Applied Soft Computing*, 13(12): p. 4750-4762.

Trisna, T., et al., (2016). Multi-objective optimization for supply chain management problem: A literature review. *Decision Science Letters*, 5(2): p. 283-316.

This article can be cited: Sahebjamnia, N. Goodarzian, F. & Hajiaghaei-Keshteli, M. (2020). Optimization of Multi-period Three-echelon Citrus Supply Chain Problem. *Journal of Optimization in Industrial Engineering*. 13 (1), 39-53.

http://www.qjie.ir/article_538019.html
DOI: 10.22094/JOIE.2017.728.1463