J Intell Manuf (2014) 25:699–726
DOI 10.1007/s10845-012-0714-x

A polar-based guided multi-objective evolutionary algorithm to search for optimal solutions interested by decision-makers in a logistics network design problem

Hossein Rajabalipour Cheshmehgaz · Md. Nazrul Islam · Mohammad Ishak Desa

Received: 14 March 2012 / Accepted: 7 November 2012 / Published online: 28 November 2012

Abstract In practical multi-objective optimization problems, respective decision-makers might be interested in some optimal solutions that have objective values closer to their specified values. Guided multi-objective evolutionary algorithms (guided MOEAs) have been significantly used to guide their evolutionary search direction toward these optimal solutions using by decision makers. However, most guided MOEAs need to be iteratively and interactively evaluated and then guided by decision-makers through re-formulating or re-weighting objectives, and it might negatively affect the algorithms performance. In this paper, a novel guided MOEA that uses a dynamic polar-based region around a particular point in objective space is proposed. Based on the region, new selection operations are designed such that the algorithm can guide the evolutionary search toward optimal solutions that are close to the particular point in objective space without the iterative and interactive efforts. The proposed guided MOEA is tested on the multi-criteria decision-making problem of flexible logistics network design with different desired points. Experimental results show that the proposed guided MOEA outperforms two most effective guided and non-guided MOEAs, R-NSGA-II and NSGA-II.

Keywords Multi-objective optimization problems · Guided multi-objective evolutionary algorithms · Polar coordinate system · Flexible logistics network design problem

In the loving memory of my mother, 'Turan Motallebizadeh' and dedicated to my beloved wife, 'Farahnaz Kazemipour' who inspired me in so many ways—by Hossein Rajabalipour C.

H. Rajabalipour Cheshmehgaz (✉)· Md. N. Islam · M. I. Desa
Universiti Teknologi Malaysia (UTM), Skudai, Johor, Malaysia
e-mail: hrajabalipour@uk.ac.ir

Introduction

In practical multiple objective optimization problems (MOPs), there are more than one optimal solution, called Pareto solutions (PSs) or Pareto Fronts (PFs). Respective decision-makers might be interested in particular optimal solutions, which have objectives value closer to specified objectives value (reference) than other solutions (Deb et al. 2006). Multi-objective evolutionary algorithms (MOEAs) have been generally used to search for all PSs or PFs; however, guided versions of MOEAs have been intentionally designed to search for these particular optimums, more effectively. The decision-maker must initially provides certain desired value of objectives (as a reference or a goal) to the guided MOEAs, and the algorithms incorporate these given information to guides the evolutionary search toward the optimal solutions that are located as close as possible to the desired values considered by him or her (Shen et al. 2010).

Most guided MOEAs rely on interactive evaluations (e.g. Gong et al. 2011; Thiele et al. 2009) and weighting methods (e.g. Deb et al. 2006; Kim et al. 2012) to adjust their search directions. The search direction is not always controllable and predictable to reach the desired optimal solutions, while there is tried-and-error strategy used with weighting methods (e.g. Ozcan and Toklu 2009). Since decision-makers cannot be sure whether better solutions are available or not, the assigned weights have to be changed frequently, and the algorithms must be re-run accordingly. The most effective guided MOEAs like R-NSGA-II (Deb et al. 2006) that do not use such interactive weights setting effort are able to find the solutions close to some predefined points (references); but, the solutions found are not as closer to optimal solutions as some solutions found by non-guided MOEAs, like NSGA-II. Actually, no decision-maker prefers a closer solution to solutions with better objectives.

In this paper, a new guided MOEA is proposed that gradually guide the evolutionary search toward the desired values, such that respective decision-makers meet best optimal solutions (near to the desired values as can as possible) without the need of interactive evolutions and re-running the algorithm.

In the rest of this paper, Second section is considered to review a literature of existing guided MOEAs, since a practical case study of the 4-echelon flexible logistics network design problem with a desired configuration is presented and formulated in the next section. A new guided MOEA is proposed in fourth section, and all comparison results are illustrated and discussed in fifth section. Finally, the conclusion are given in last section.

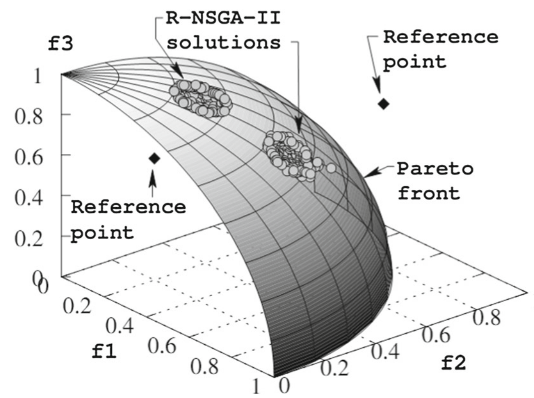


Fig. 1 Non-dominated solutions found by R-NSGA-II - near to two reference points and PFs on DTLZ2 (Deb et al. 2006)

Literature review

The first guided evolutionary algorithm was proposed by Fonseca and Fleming (1993). They proposed a modified domination relation in order to guide population toward an appropriate goal that iteratively changed and reformulated. With the updated formulation, the objectives, which are not yet satisfying the goal, are specified by decision-maker(s), and these objectives are given higher importance for next generations. Although the next version of their algorithm was proposed later (Fonseca and Fleming 1998), two issues were always reported with these algorithms: the limited capacity of human cognition in evaluation and the weak ability of searching for solutions near to sparticular or fixed points in objective space (Rachmawati and Srinivasan 2010).

Cvetkovic and Parmee (2002a, 2002b; Parmee et al. 2001, 2000) used fuzzy preference into weights and defined a new formula of domination through assigning the fuzzy-based weights to objectives. However, the fuzzy-based definition is not able to indicate "how much" one solution is better. Consequently, it causes a very coarse guidance and hardly controllable search toward particular points in objective space. In similar work, Jin and Sendhoff (2002) used interval values for weights instead of single values, while the weights are still made by the fuzzy preference.

Branke et al. (2000; Branke et al. 2001) proposed a new ranking method (system) through changing the shape of dominated regions by solutions in population. Although the algorithm is a guided MOEA, it employs multiple guided searches for all optimal solutions. The algorithm cannot be used to search for particular optimal solutions.

Tan et al. (2003) employed multiple goals with two types of priority, hard and soft, provided by decision-maker(s). A goal-sequence domination scheme is considered to allow the hard and soft priorities for better decision of user in optimization. However, the algorithm relies on an interactive optimization, and the quality of final solutions is significantly dependant on the sequence and priority specified by decision-maker(s).

Deb et al. (2006) have proposed a most-used algorithm in the present area: R-NSGA-II. The algorithm is a guided version of a non-guided version of its corresponding algorithm, NSGA-II (Deb et al. 2002) through replacing the crowing distance - that is used to keep the diversity along the PFs - with ranked distance. The ranked distance helps the algorithm to improve the solutions - in population - toward the multiple predefined points, namely reference points, in objective space. Figure 1 shows 3D-objective space of DTLZ2 problem and some non-dominated solutions found by R-NSGA-II after numbers of generation (iteration). The solutions are located as close as possible not only to the reference points, but also to PFs (optimal solutions).

Zitzler and Kunzli (2004; Zitzler et al. 2008) defined some indicator-based relationships based on hypervolume measure in order to guide the evolutionary search to particular points in objective space. However, the calculation of hypervolume is noticeably time-consumed in their proposed guided algorithm, SPAM.

In more-recent work, Kim et al. (2012) have proposed a preference-based solution selection algorithm (PSSA) in which a preferred solution can be selected from all non-dominated solutions based on partial and global evaluations. User's preference for the objectives is represented by a degree of consideration using fuzzy measures. The proposed algorithm shows better performance on many-objective optimization problems as compared with some other MOEAs like NSGA-II. It is due to emphasizing on specific objectives by the degree of consideration that it is much better effective when the algorithms deal with MOPs with many objectives. However, the definition of fuzzy measures needs an expert(s) to be defined appropriately.

All aforementioned works use different approaches to guide the evolutionary search to particular optimal solutions, which can nearly satisfy specific goals. Some interactively use decision-makers' preference to change the domination definition or adapt the respective weights (of objectives), whereas others change the definition once in order to find some desired solutions. However, the main challenge in guided MOEAs that use interactive guidance arise from the difficulty of specifying a suitable change in definition or new weights for objectives. The trade-off efforts must be taken into account for respective decision-makers to decide on the best setting or changes.

4-echelon flexible logistics network design problem

There has been a need for low-cost shipment of products, from suppliers to customers. Many companies have been trying to build up, or even redesign their logistics networks to eliminate unnecessary delays in their shipment, while at the same time, to reduce associational costs or at least not to exceed a particular amount (Timm and Herbert 2009).

In general, a logistics network consists of a set of facilities and configurations, which determine product shipments between the facilities. Logistics network design problem is concerned with how customer demands or service requirements should be treated such that the shipment time and particular costs, are minimized, simultaneously (Vidal and Goetschalckx 1997).

Meanwhile, the success of Dell Company with its cost-effective logistics networks, has influenced other companies' long-term strategy to benefit from the useful flexibilities in their logistics networks (Gen et al. 2008). The networks are more flexible with facility skipping through extra potentially direct shipments between facilities and customers. These logistics networks are more effective with lower transportation cost, delivery time, and consequently, increasing customer satisfaction. However, the trade-off between direct and indirect shipments is still a critical issue in decision-making with these networks; while the decision of opening or closing the facilities makes the decision more complex.

Meanwhile, three criteria of transportation cost, facility cost and delivery time have been more considered and addressed as quantitative and qualitative criteria. Therefore, any long-term decision on the flexibility used in logistics network design needs to deal with these criteria, carefully and wisely. These criteria are briefly explained as follows: (Rajabalipour Cheshmehgaz et al. 2011)

Facility Cost: A long-term facility decision is concerned with determining the number of facilities involved and their location. The problem was first recognized by Cooper (1963) as the multi-Weber's problem. In this research, the problem deals with the decision of opening or closing the potential facilities in a period with two sub costs: setup cost and providing/inventory (or supplying/inventory) cost. Setup cost is any cost of new contracts with suppliers, opening/maintaining facilities and etc. where initial budget is often required (Park et al. 2010). Providing/inventory cost is the cost per unit of products in which the units are produced and stored (or maintained) until their shipment to other facilities or customers (Suo et al. 2011).

Transportation Cost: A well-designed transportation network can effectively reduce cost and improve the quality of service (Gen et al. 2006). Transportation problem was first introduced by Hitchcock (1941) looking for the optimum way of transporting homogenous products from some sources to some destinations. In this research, the transportation cost includes the cost of transporting good from facilities to customers.

Delivery Time: the delivery time has recently become a challenge for many logistic companies. Since Stalk and Hout (1990) introduced the notion on time compression, the problem about the time in logistics network has been researched by many researchers (Yang et al. 2007). The time depends on many factors of customer-supplier pairs, including physical distance, transportation mode (e.g. Train, ship, road and, etc.) and also inventory control model in some facilities (Park et al. 2010).

Objectives formulation

In this paper, a wide-used logistics network, 4-echelon flexible Logistics network (4-fLN) with single source supplying strategy (Gen et al. 2008) is considered, and their objectives are formulated. The network comprises of potential suppliers (S), distribution centres (DC), retailers (R) and actual customers (C) at the first level. Each customer has pre-specified demand of single item product for a period (e.g. season, year, etc.). The main objective of the problem is to calculate the status (decision on being open or close) of suppliers, distribution centres or retailers; and transportation routes in order to minimize the delivery time, transportation cost and facility costs, simultaneously. With consideration that the network can be flexible which allows potential (probably more expensive) and direct shipments from suppliers to DCs, suppliers to retailers, suppliers to customers, DCs to retailers and DCs to customers (see Fig. 2). According to many real logistics networks, the facilities incapacitated (Rajabalipour Cheshmehgaz et al. 2011), thus this research considers this type of facility.

Some notation is now introduced for mathematical formulations of the objectives as follows:

Facility Cost Objective: the facility cost is calculated with three sub-costs in the facilities of retailers, DCs and suppliers, individually, by Eqs. (1), (2) and (3). The descriptions below to each expression in the equations indicate, each cost including setup, inventory and providing costs in

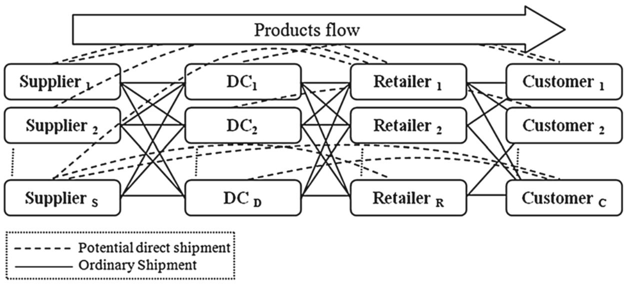


Fig. 2 4-echelon flexible Logistics network with potential direct shipments

Indices

s	Index of suppliers $(1..S)$
d	Index of distribution centers $(1..D)$
r	Index of retailers $(1..R)$
c	Index of customer $(1..C)$

Parameters

Costs in facilities	
SC_s^S	Fixed setup cost for supplier s
SC_d^D	Fixed setup cost for DC d
SC_r^R	Fixed setup cost for retailer r
PC_s	Per unite supplying (providing) Cost by supplier s
IC_d^D	Per unite inventory cost in DC d
IC_r^R	Per unite inventory cost in retailer r
Costs in transportation	
$TC_{s,d}^{SD}$	Per unit transportation Cost between supplier s and DC d
$TC_{s,r}^{SR}$	Per unit transportation Cost between supplier s and retailer r
$TC_{s,c}^{SC}$	Per unit transportation Cost between supplier s and customer c
$TC_{d,r}^{DR}$	Per unit transportation Cost between DC d and retailer r
$TC_{d,c}^{DC}$	Per unit transportation Cost between DC d and customer c
$TC_{r,c}^{RC}$	Per unit transportation Cost between retailer r and customer c
Delivery times	
$T_{s,d}^{SD}$	Shipment time between supplier s and DC d
$T_{s,r}^{SR}$	Shipment time between supplier s and retailer r
$T_{s,c}^{SC}$	Shipment time between supplier s and customer c
$T_{d,r}^{DR}$	Shipment time between DC d and retailer r
$T_{d,c}^{DC}$	Shipment time between DC d and customer c
$T_{r,c}^{RC}$	Shipment time between retailer r and customer c
D_c	Demand of customer c

Decision variables

$CR_{c,r} \in \{0, 1\}$	Whether or not to assign customer c to retailer r
$CD_{c,d} \in \{0, 1\}$	Whether or not to assign consumer c to DC d
$CS_{c,s} \in \{0, 1\}$	Whether or not to assign consumer c to supplier s
$RD_{r,d} \in \{0, 1\}$	Whether or not to assign retailer $r$to DC d
$RS_{r,s} \in \{0, 1\}$	Whether or not to assign retailer r to supplier s
$DS_{d,s} \in \{0, 1\}$	Whether or not to assign DC $d$to supplier s
Derivative variables	
$SSt_s \in \{0, 1\}$	0/1 status of Suppliers (open or close) calculated by Eq. (10)
$DSt_d \in \{0, 1\}$	0/1 status of DC d (open or close) calculated by Eq. (11)
$RSt_r \in \{0, 1\}$	0/1 status of retailer r(open or close) calculated by Eq. (12)

the facilities. Finally, facility cost objective are formulated in Eq. (4) that minimized the all mentioned costs in facilities.

$$TotalFacilityCost_{retailers} = \sum_{r=1}^{R} \left[\underbrace{SC_r^R \times RSt_r}_{\substack{\text{Setup cost of Retailer } r \\ (if\ it\ is\ open)}} + \underbrace{IC_r^R \times \sum_{c=1}^{C} D_c \times CR_{c,r}}_{\substack{\text{Inventory cost in Retailer } r \\ (if\ it\ is\ open)}} \right] \tag{1}$$

$$TotalFacilityCost_{DCs} = \sum_{d=1}^{D} \left[\underbrace{SC_d^D \times DSt_d}_{\substack{\text{Setup cost of DC } d \\ (if\ it\ is\ open)}} + \underbrace{IC_d^D \times \sum_{c=1}^{C} \left[\underbrace{D_c \times CD_{c,d}}_{\substack{\text{Direct demands} \\ (Customer\ to\ DC)}} + \sum_{r=1}^{R} \underbrace{D_c \times CR_{c,r} \times RD_{r,d}}_{\substack{\text{Indirect demands} \\ (Customer\ to\ Retailer\ to\ DC)}} \right]}_{\substack{\text{Inventory cost in DC } d \\ \text{(if it is open)}}} \right] \tag{2}$$

$$TotalFacility\,Cost_{Suppliers} = \sum_{s=1}^{S} \Bigg[\underbrace{SC_s^S \times SSt_s}_{\substack{\text{Setup cost of Supplier } s \\ (if\ it\ is\ open)}} + \underbrace{PC_s \times \sum_{c=1}^{C} \Bigg[\underbrace{D_c \times CS_{c,s}}_{\substack{\text{Direct demands} \\ (Customer\ to\ Supplier)}} + \sum_{d=1}^{d} \Big[\underbrace{D_c \times CD_{c,d} \times DS_{d,s}}_{\substack{\text{Indirect demands} \\ (Customer\ to\ DC\ to\ Supplier)}} \Big] + \sum_{r=1}^{R} \Big[\underbrace{D_c \times CR_{c,r} \times RS_{r,s}}_{\substack{\text{Indirect demands} \\ (Customer\ to\ Retailer\ to\ Supplier)}} \Big] + \sum_{r=1}^{R} \sum_{d=1}^{D} \Big[\underbrace{D_c \times CR_{c,r} \times RD_{r,d} \times DS_{d,s}}_{\substack{\text{Indirect demands} \\ (Customer\ to\ Retailer\ to\ DC\ to\ Supplier)}} \Big] \Bigg]}_{\substack{\text{Supplying cost in Supplier } s \\ (if\ it\ is\ open)}} \Bigg] \tag{3}$$

$$\begin{aligned} Minimizing(&FacilityCost \\ &= TotalFacilityCost_{retailers} \\ &\quad + TotalFacilityCost_{DCs} \\ &\quad + TotalFacilityCost_{Suppliers}) \end{aligned} \tag{4}$$

Transportation Cost Objective: this cost is calculated with three sub-costs with the shipments from suppliers, DCs and retailers to their front echelons. The transportation costs of each of these shipments are formulated by Eqs. (5), (6) and (7). The cost of all goods transportation from suppliers to any possible DC, retailer and customer by direct and indirect shipments is formulated by four individual summations in Eq. (5). The same formulation for the possible shipments from DCs and retailers are presented by Eq. (6) and (7), respectively. Finally, the total transportation cost is formulated and the related objective in introduced in Eq. (8).

$$\begin{aligned} &TransportationCost_{Suppliers} \\ &= \sum_{s=1}^{S} \Bigg[\sum_{c=1}^{C} \Bigg[\underbrace{TC_{s,c}^{SC} \times CS_{c,s} \times D_c}_{\substack{\text{Transportation cost by direct shipment} \\ (Supplier\ to\ Customer)}} \Bigg] \\ &+ \sum_{c=1}^{C} \sum_{r=1}^{R} \Bigg[\underbrace{TC_{s,r}^{SR} \times RS_{r,s} \times CR_{c,r} \times D_c}_{\substack{\text{Transportation cost by indirect shipment} \\ (Supplier\ to\ Retailer\ to\ Customer)}} \Bigg] \\ &+ \sum_{c=1}^{C} \sum_{d=1}^{D} \Bigg[\underbrace{TC_{s,d}^{SD} \times DS_{d,s} \times CD_{c,d} \times D_C}_{\substack{\text{Transportation cost by indirect shipment} \\ (Supplier\ to\ DC\ to\ Customer)}} \Bigg] \\ &+ \sum_{c=1}^{C} \sum_{d=1}^{D} \sum_{r=1}^{R} \Bigg[\underbrace{TC_{s,d}^{SD} \times DS_{d,s} \times RD_{r,d} \times CR_{c,r} \times D_C}_{\substack{\text{Transportation cost by indirect shipment} \\ (Supplier\ to\ DC\ to\ Retailer\ to\ Customer)}} \Bigg] \Bigg] \end{aligned} \tag{5}$$

$$\begin{aligned} &TransportationCost_{DCs} \\ &= \sum_{d=1}^{D} \Bigg[\sum_{c=1}^{C} \Bigg[\underbrace{TC_{d,c}^{DC} \times CD_{c,s} \times D_c}_{\substack{\text{Transportation cost by direct shipment} \\ (DC\ to\ Customer)}} \Bigg] \\ &+ \sum_{c=1}^{C} \sum_{r=1}^{R} \Bigg[\underbrace{TC_{d,r}^{DR} \times RD_{r,d} \times CR_{c,r} \times D_c}_{\substack{\text{Transportation cost by indirect shipment} \\ (DC\ to\ Retailer\ to\ Customer)}} \Bigg] \Bigg] \end{aligned} \tag{6}$$

$$\begin{aligned} &TransportationCost_{Retailers} \\ &= \sum_{r=1}^{R} \sum_{c=1}^{C} \left[TC_{r,c}^{RC} \times CR_{c,r} \times D_c \right] \end{aligned} \tag{7}$$

$$Minimizing(TransportationCost = \sum_{i\in\{Suppliers, DCs, Retailers\}} TransportationCost_i) \tag{8}$$

Delivery Time Objective: the maximum delivery time in the network can be calculated within Eq. (9), while the related objective is formulated as a minimizing objective.

$$\begin{aligned} &Minimizing(MaximalDeliveryTime \\ &= \max\left\{\left(T^{SD}_{s,d} + T^{DR}_{d,r} + T^{RC}_{r,c}\right) \times DS_{d,s} \times RD_{r,d} \times CR_{c,r},\right. \\ &\left.\left(T^{SR}_{s,r} + T^{RC}_{r,c}\right) \times RS_{r,s} \times CR_{c,r}, T^{SC}_{s,c} \times CS_{s,s}\right\}_{\forall s,d,r,c} \end{aligned} \tag{9}$$

Some decision variables have been used only in the mentioned equations to make them traceable and more understandable. These variables are given the values by other decision variables; while the values indicate whether the facility is open or not (closed) - see Eqs. (10), (11) and (12).

$$SSt_s = \begin{cases} 1 & \text{if } S_s > 0 \\ 0 & \text{otherwise} \end{cases}, \forall s$$

$$S_s = \sum_{c=1}^{C}\sum_{d=1}^{D}\sum_{r=1}^{R}(CS_{c,s} + CR_{c,r} \times RS_{r,s} + CD_{c,d} \times DS_{d,s} + CR_{c,r} \times RD_{r,d} \times DS_{d,s}) \tag{10}$$

$$DSt_d = \begin{cases} 1 & \text{if } D_d > 0 \\ 0 & \text{otherwise} \end{cases}, \forall d \tag{11}$$

$$D_d = \sum_{c=1}^{C}\sum_{r=1}^{R}(CD_{c,d} + CR_{c,r} \times RD_{r,d})$$

$$RSt_r = \begin{cases} 1 & \text{if } \sum_{c=1}^{C} CR_{c,r} > 0 \\ 0 & \text{otherwise} \end{cases}, \forall r \tag{12}$$

The objectives are subject to some constraints as follows:

Constraint (13) secures the single source assumption that each Customer^i must be supplied by only one facility. Constraint (14) means that if a retailer is open, it must be supplied by only one facility (supplier or DC). In addition, Constraint (15) describes the same constraint with DCs that there will be only supplied by only one supplier according to single source assumption.

$$\sum_{r=1}^{R} CR_{c,r} + \sum_{d=1}^{D} CD_{c,d} + \sum_{s=1}^{S} CS_{c,s} = 1, \forall c \tag{13}$$

$$\sum_{d=1}^{D} RD_{r,d} + \sum_{s=1}^{S} RS_{r,s} = 1, \forall r, if\ St_r = 1 \tag{14}$$

$$\sum_{s=1}^{S} DS_{d,s} = 1, \forall d, if\ DSt_d = 1 \tag{15}$$

In addition to the objectives defined, the logistics networks have been enhanced through considering the flexibility of shipment over the whole networks. With this flexibility, there are many optimal configurations with the variety of optimal costs and time. Nonetheless, only a few of these optimal configurations can be interested by respective diction-makers. These might be such budget limitations for transportation and facility along with delivery times, thus a few of these configurations can be satisfied or at leas as close as possible to these limitations. Therefore, the interested values (as soft constraints) can be considered as a desired point or a goal point in objective space that can be used to indicate how close these optimal solutions are to this point.

The proposed guided multi-objective evolutionary algorithm

In this section, a new guided MOEA, GMOEA is introduced, and the related steps are explained with some illustrative examples. First, a dynamic polar-based region around the particular point, called VIP region, is defined. Afterward, the procedure of dynamic change of the region toward a desire point, goal is explained. An ordering system that gives unique order numbers to the current solutions in population is proposed. The order numbers are regularly changed, such that the evolutionary search can be guided toward PFs, which are as close as possible to the particular point.

Region assignment technique

In order to search for closer optimal solutions to a particular point in objective space, or optimistically better than the point but nearby, it is important to indicate how the search region is bounded. In some similar efforts (but not for the similar purpose), the boundary is considered as a hard constraint to the search algorithms, and any solutions outside the region must be eliminated, or at least, must be considered with lower worth or fitness (e.g. Cagnina et al. 2011; da Silva et al. 2011. The boundary, here, is considered as a soft constrain in which it is not necessary to maintain only the solutions in populations that are located inside the boundary (or region).

In the present work, only the optimization problems with positive and minimizing objectives are considered. However, any other optimization problems can be easily changed to the considered problems according to the way that has been recently proposed by Rajabalipour Cheshmehgaz et al. (2012a).

Initial region and its dynamic contracting in 3d space

Evolutionary algorithms rely on a variety of solutions with different genotype in its population to search effectively and efficiently. It has been proven that these algorithms might converge to local optimal solutions instead of global ones, if

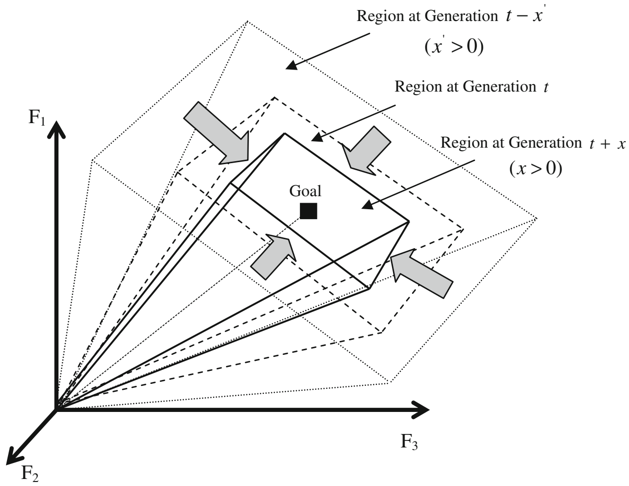


Fig. 3 3D-Objective space and contracting a polar-based region around a desired point, goal

the algorithms encounter more limitation with a low range of diversity in ssolutions, particularly in early generations (Sato et al. 2007; Zhang and Li 2007). Therefore, the proposed algorithm, GMOEA is given a larger polar-based region around a particular point called goal afterward. The algorithm explores and exploits the region, while the region gradually becomes smaller during further generations (see Fig. 3).

The region that is considered in the figure is specified by four angles in 3D objective space for 3-objective optimization problems (it can be extended for more objectives). The angels are updated in each generation to make the region more contracted toward the goal.

Consider a goal in the polar coordinate system, Goal $= (r, \theta_1, \theta_2)$ in 3D objective space with three axioms of F_1, F_2 and F_3 (as three objective functions). Fig. 4 illustrates the four angles on two different planes and their changes in two different generations. θ_1^l and θ_1^u are, respectively, lower and upper angles of the region on the plane that passes through F_2 and F_3. In addition, θ_2^l and θ_2^u are, respectively, lower and upper angles of the region on the plane passing though F_1 and $F_{2,3}$ which is shadow vector of the goal on another plane passing through F_2 and F_3.

The upper and lower angles become smaller and larger, respectively, with different magnitudes in order to create a smaller region around the goal. In the figure, the angles in initial state (before generation) and after Generation t are shown in the left side and the right side, relatively.

Assuming that the initial population is available, the proposed dynamic change of the angles is formulated as follows:

The initial angles of the region are set by the equations of corresponding angles of current solutions in population according to Eqs. (16), (17), (18) and (19). In the cases of having under or over values with any angle ($> \pi/2$ or < 0), the initial value is set to zero.

Parameter	Description
$IntPop$	Initial population, randomly generated
N	Population size
t_0	Initial state, before starting evolutionary generation
t	t th generation
$G = [g_1, g_2, g_3]$	Cartesian coordinate-based Goal in objective space
$G = [r^g, \theta_1^g, \theta_2^g]$	Polar coordinate-based Goal according to Eq. (1)
$\left[\theta_1^l(t_0), \theta_1^u(t_0)\right]$	Initial lower and upper bound of first angle of the polar-based region around the goal point in objective space
$\left[\theta_2^l(t_0), \theta_2^u(t_0)\right]$	Initial lower and upper bound of second angle of the polar-based region around the goal point in objective space
$\left[\theta_1^l(t), \theta_1^u(t)\right]$	lower and upper bound of first angle of the polar-based region around the goal point in objective space at Generation t
$\left[\theta_2^l(t), \theta_2^u(t)\right]$	Initial lower and upper bound of second angle of the polar-based region around the goal point in objective space at Generation t
$\Delta\theta_1^l$	Magnitude of increase with θ_1^l at each generation
$\Delta\theta_1^u$	Magnitude of decrease with θ_1^u at each generation
$\Delta\theta_2^l$	Magnitude of increase with θ_2^l at each generation
$\Delta\theta_2^u$	Magnitude of decrease with θ_2^u at each generation
$\theta_1(n)$	First angle of solution $n, (n = 1, \ldots, N)$in polar coordinate system
$\theta_2(n)$	Second angle of solution $n, (n = 1, \ldots, N)$ in polar coordinate system
NG	Number of generations

$$\theta_1^l(t_0) = \begin{cases} \theta_1^g - \left[\sum_{i=1}^N \theta_1(i)/N\right] & \text{if } \geq 0 \\ 0 & \text{othwerwise} \end{cases} \tag{16}$$

$$\theta_1^u(t_0) = \begin{cases} \theta_1^g + \left[\sum_{i=1}^N \theta_1(i)/N\right] & if \leq \pi/2 \\ \pi/2 & \text{othwerwise} \end{cases} \tag{17}$$

$$\theta_2^l(t_0) = \begin{cases} \theta_2^g - \left[\sum_{i=1}^N \theta_2(i)/N\right] & \text{if } \geq 0 \\ 0 & \text{othwerwise} \end{cases} \tag{18}$$

$$\theta_2^u(t_0) = \begin{cases} \theta_2^g + \left[\sum_{i=1}^N \theta_2(i)/N\right] & \text{if } \leq \pi/2 \\ \pi/2 & \text{othwerwise} \end{cases} \tag{19}$$

The magnitude of change with each angle is formulated by Eqs. (20), (21), (22) and (23). Obviously, each angle may be changed (expanded or contracted) with different magnitude during the generations. All angles have nearly same values

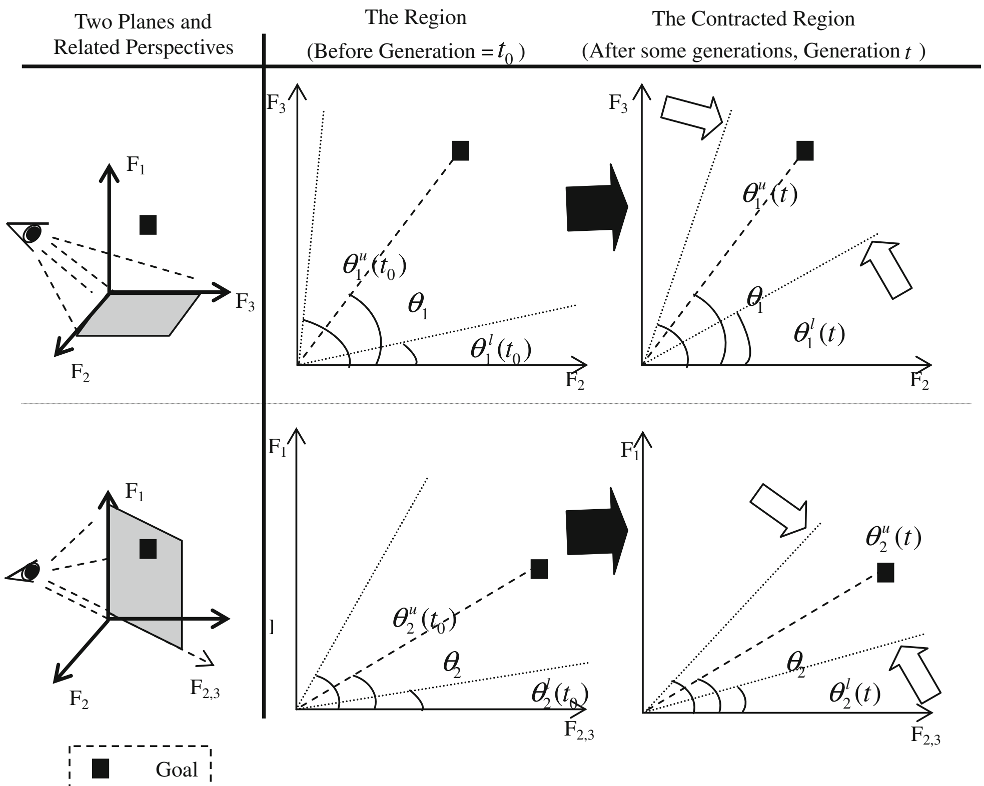


Fig. 4 Region contracting along with generation and its related angles

at the final generation. In other words, the region will be very small at the end.

$$\Delta\theta_1^l = \left|\theta_1^g - \theta_1^l(t_0)\right| / NG \tag{20}$$

$$\Delta\theta_1^u = \left|\theta_1^g - \theta_1^u(t_0)\right| / NG \tag{21}$$

$$\Delta\theta_2^l = \left|\theta_2^g - \theta_2^l(t_0)\right| / NG \tag{22}$$

$$\Delta\theta_2^u = \left|\theta_2^g - \theta_2^u(t_0)\right| / NG \tag{23}$$

Finally, the new values of the angles at Generation t are calculated by Eqs. (24), (25), (26) and (27). The upper and lower angles of region are contacted and expanded, respectively.

$$\theta_1^l(t) = \theta_1^l(t-1) + \Delta\theta_1^l \tag{24}$$

$$\theta_1^u(t) = \theta_1^u(t-1) - \Delta\theta_1^u \tag{25}$$

$$\theta_2^l(t) = \theta_2^l(t-1) + \Delta\theta_2^l \tag{26}$$

$$\theta_2^u(t) = \theta_2^u(t-1) - \Delta\theta_2^u \tag{27}$$

Ordering system

Mate selection, crossover, mutation, environmental selection (or elitism strategy), are essential operations in MOEAs. The first three operations generate new solutions (offspring), while the environmental selection operation selects some solutions from the previous solutions (parents) and new generated solutions to make a new population (Deb 2001). Surly, ordering of solutions (individuals) has a significant influence on the selection operations, and as a consequence, on the algorithm's performance. Because the fitness of solutions that indicates their goodness is calculated based on their order numbers, and selection operations (mate and environmental) in MOEAs use the numbers to create next populations. Most popular ordering system employs the domination concept (Deb et al. 2002; Zitzler et al. 2001) that is also considered in the proposed ordering system.

The proposed system is similar to an ordering system that proposed Rajabalipour Cheshmehgaz et al. (2012b), but the current solutions, including parents and offspring, are divided into two groups instead of three groups. The solutions in the first group are definitely preferred, so then the order numbers of these solutions are lower than the solutions in the second group. However, each group needs a local ordering system.

First group: All first Fronts (known-PFs) inside the region are in this group, and they will be given lower numbers (frontier) comparing with another group's members. Furthermore, the members in this group will be locally ordered based on Crowding Distance Sorting method proposed by (Deb et al. 2002) in order to maintain the required diversity around the goal pint.

A simple 2D-objective space with eight solutions (in a population) is illustrated in Fig. 5 to describe the proposed

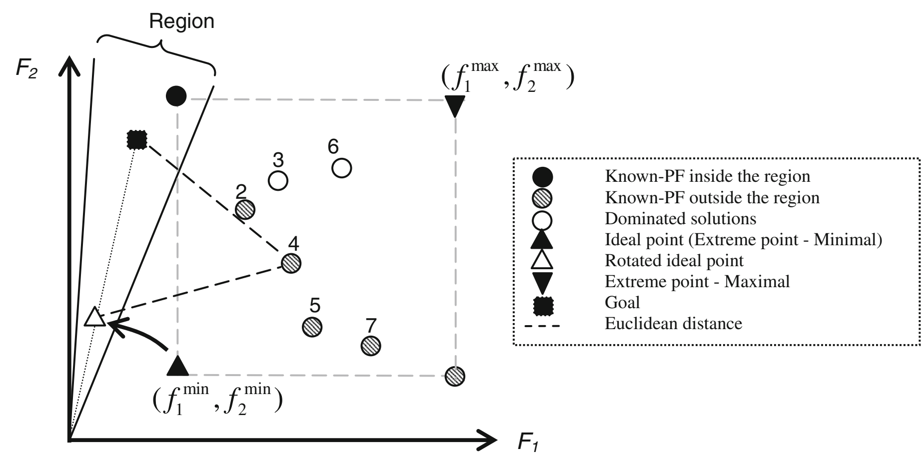


Fig. 5 2D objective space and the polar-based region—ordering the current solutions

ordering system. In the figure, only Solution 1 is inside the region, then it is the only member of the first group and is given number one.

Second group: the other solutions are placed in this group, and are given higher-order numbers (or lower rank) as compared with the first group. Each solution should be locally ordered based on two normalized Euclidean distances to the goal, and the current ideal point which can be easily indicated with the minimal value of the objectives of current solutions (in population). The ideal point might be outside the region; therefore, in order to guide the search not far from the goal, the point is rotated based on the original point into the middle of the region as it is shown in Fig. 5. The figure also illustrates the order number of solutions, which are in the second group, whether are dominated or not. These solutions are ordered based on the lowest value of D formulated by Eq. 28.

$$D = \sqrt{\sum_{i=1}^{ObjN} \frac{F_i(x) - G(i)}{f_i^{\max} - f_i^{\min}}} + \sqrt{\sum_{i=1}^{ObjN} \frac{F_i(x) - Ideal(i)}{f_i^{\max} - f_i^{\min}}} \tag{28}$$

where

$ObjN$: Number of objectives
$f_i^{\min}$: Minimal value of i^{th} objective of current solutions
$f_i^{\max}$: Maximal value of i^{th} objective of current solutions
G: goal in objective space
x: Solution (parent or offspring)
F_i: Objective i

Mate selection

The first and important evolutionary operation is the mate selection operation that usually works based on a particular ordering system. We use the binary tournament selection operation that has been proposed by Rajabalipour Cheshmehgaz et al. (2012b); but the operation here works based on the proposed ordering system. With this selection, the decision in selection is made based on whether the selected solutions are inside the VIP region or not.

VIP environmental selection

In each generation, the offspring generated from the parents can be involved in creating the next population through two main strategies: $(\mu + \lambda)$-ES and (μ, λ)-ES (Engelbrecht 2007). In the first strategy, λ solutions are selected from the union set of parents and offspring, whereas in the second strategy, only μ solutions form λ offspring are chosen as new population for next generations. To direct the evolutionary search to particular optimal solution, we consider the first strategy to do environmental selection in the proposed guided MOEA. The proposed operation, VIP Environmental Selection (VIP-ES) operation is used to select the current solutions based their order numbers that calculated through the proposed ordering systems.

Steps of the proposed guided MOEA

The steps of the proposed guided MOEA are shown in Fig. 6. Initial population is randomly generated. The initial polar-based region around the goal is established by Eqs. (16), (17), (18) and (19). In the following steps, iteratively, new populations are generated, while the region is adjusted (contracted toward the goal, gradually) after each generation based on Eqs. (24), (25), (26) and (27). Meanwhile, the evolutionary computing benefits from two new selections designed based on the ordering system aforementioned.

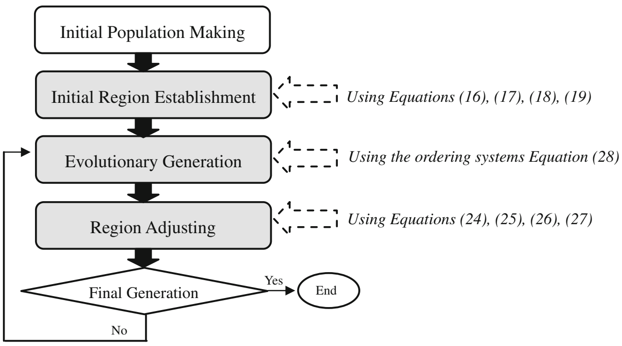


Fig. 6 The steps of the proposed guided MOEA

Reference non-dominated sorting genetic algorithm II (R-NSGA-II)

Reference Non-dominated Sorting Genetic Algorithm II (R-NSGA-II) is a guided MOEA that was proposed by Deb et al. (2006). They combined a preference-based strategy within the ordering systems and environmental selection used in NSGA-II (Deb et al. 2002) to search for a set of solutions near the reference points predefined by a decision-maker. The algorithm uses a weighted Euclidean distance (Eq. 29) to order (or rank) the solutions in population based on their distance to the reference points in objective space. The algorithm additionally benefits from a particular neighborhood to maintain the diversity among the solutions near the reference-point. The weighted Euclidean distance is given by:

$$d_{ij} = \sqrt{\sum_{i=1}^{M} w_i \left(\frac{f_i(\mathbf{x}) - z_i}{f_i^{\max} - f_i^{\min}} \right)^2} \tag{29}$$

where $f_i^{\max}$ and $f_i^{\min}$ are the population maximum and minimum function values of i^{th} objective, $w_i = 1/M$ (M: number of objectives), and z_i is the value of i^{th} objective of the reference point.

The ordering systems and environmental selection operations are the main effective parts of any MOEA. Here, the three following steps are replaced to the original ordering system and environmental selection in NSGA-II to incorporate two ideas of: (1) emphasizing more on the solutions closer to the point, and (2) de-emphasizing on the solutions within the neighborhood to maintain the considered diversity near to each reference point:

Step 1: The solutions in population are sorted based on their distance from the points based on Eq. 29.

Step 2: After evolutionary computing (crossover, mutation and, etc.) the nearest solutions to the preference points are preferred in a tournament selection (as environmental selection) and in forming the new population from the combined population of parents and offspring.

Step 3: To maintain the diversity of solutions around the goal, the environmental selection, first, picks a random solution from the current non-dominated in population. Thereafter, all solutions having a sum of normalized difference (calculated by Eq. 29) in objective values less than the chosen solution are assigned an artificial large preference distance (relatively) to discourage them to remain in the race. This way, only one solution within a neighborhood is emphasized. Then, another solution from the non-dominated set (and is not already considered earlier) is picked and the above procedure is performed.

R-NSGA-II and its original one, NSGA-II along with the proposed guided MOEA were used to solve the industrial combinatorial MOPs of flexible logistics network design and assembly line balancing. The algorithms independently search for the optimal configurations for the networks and assembly lines in which some related criteria value resulted by the configurations can be as less (or at least close) as possible to the values, which are desired by respective decision-makers.

In the following sections, the experiments of running the three algorithms on the logistics network design problem with different goals are conducted, and their results are presented and discussed.

Experimental results

The logistics networks must be configured with an optimal solution (configuration), such that the respective objectives are minimized, simultaneously. The respective decision-maker prefers optimal configurations in which the facility cost, transportation cost and delivery time given by the configuration is closer to a particular point in objective space than other optimal configurations.

The distance between the solutions and the particular point in objective space, called goal, can be a criterion in the guided optimization that must be simultaneously minimized along with other objectives. Although the distance can be easily calculated via the Euclidean Distance, the respective objectives might have different ranges and scales. The normalized Euclidean distance (NED) was defined to evaluate the distance.

Figure 7 draws a picture of what solutions in population are more preferred by the decision-makers, who predefine a goal in objective space. Nine solutions and a goal are shown in the respective 2D-objective space. Suppose that only two best solutions must be selected as close as possible to the goal (A), Solutions 1 and 2 are preferred due to being known-PFs in the population, first, and then being closer to the goal than other known-PFs. Although Solution 3 is closer than these

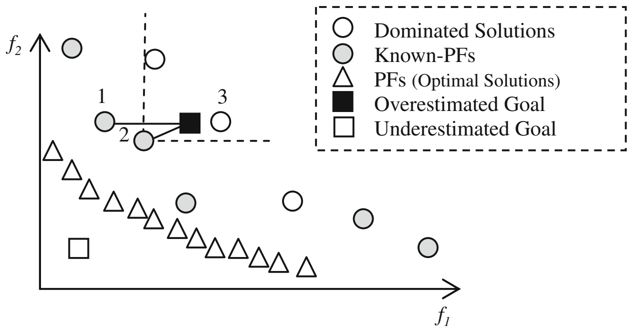


Fig. 7 Two preferred solutions around a goal in objective space

selected solutions to the goal, this solution is not preferred to any known-PFs in population, since Solution 2 dominates it.

Since decision-makers have to estimate goals at the first place, two situations can be occurred: possibly overestimated or underestimated goals. In the first situation, the respective decision-maker overestimates the goal, probably, with no positive thought, whereas the goal is underestimated in the second situation due to optimistic considerations. There is at least one feasible solutions in objective space that dominates the overestimated goal, but there is no like this solution for the underestimated goal. For instance, Goal A in Fig. 7 is an overestimated goal because Solution 2 (as feasible solution) dominates the goal. In addition, Goal B is underestimated goal, because no feasible solution (whether is PFs or not) dominates the goal.

Since any decision-maker can predefine any goal with any interest, one of the situations will be possibly occurred without notice in advance. Although, the proposed guided algorithms must search for the proper solutions without knowing about the situations, the knowledge of the situations is necessary to be given prior any evaluation and comparison study.

In the following subsections, the proposed guided MOEA, GMOEA, R-NSGA-II and NSGA-II were used to solve a flexible logistics network (fLN) with overestimated and underestimated goals. In addition, two more goals were also considered to evaluate the algorithms, while they deal with extremely underestimated and overestimated goals.

The general information about the problem is presented in Table 1. Meanwhile, all algorithms use the same crossover and mutation operations according to the information presented in Table 2, and the size of populations is fixed to 20 solutions (or individuals).

Testing with underestimated goal

GMOEA, R-NSGA-II and NSGA-II were used to search for the nearest optimal solutions to the underestimated goal with desired transportation cost (=20,000), facility cost (=20,000) and maximal delivery time (=50) for the fLN problem. All algorithms were individually run with 500 iterations (generations), and the known-PFs found by each algorithm are shown in Fig. 8. In addition, The NED of the nearest known-PF to the goals, the average of NED of all the known-PFs, the number of known-PFs and the quality of the three nearest known-PFs found by algorithms are presented in the metrics value sector in the figure. The quality of these solutions, here, is identified based on their ability of dominating or being dominated by two other solutions.

The result shows that GMOEA found a closer solution to the goal with its NED = 0.58 compared to other corresponding solutions found by other algorithms (NED values are 1.37 in R-NSGA-II, and 1.66 in NSGA-II). The solution also dominates the solution given by R-NSGA-II, while is not dominated by the corresponding solution given by NSGA-II (see the metrics value section in Fig. 8).

In addition, the average of NED of all known-PFs found by GMOEA is 0.84 which is less than the corresponding values in R-NSGA-II (=1.38) and NSGA-II (=1.78). According to the number of known-PFs, GMOEA found 20 solutions, and it is equal to the number of solutions given by NSGA-II, and more than seven solutions found by R-NSGA-II. Three sets of known-PFs that are found by R-NSGA-II, NSGA-II and GMOEA are presented in Tables 3, 4 and 5, respectively; and the nearest known-PF in each set is highlighted (in gray colour).

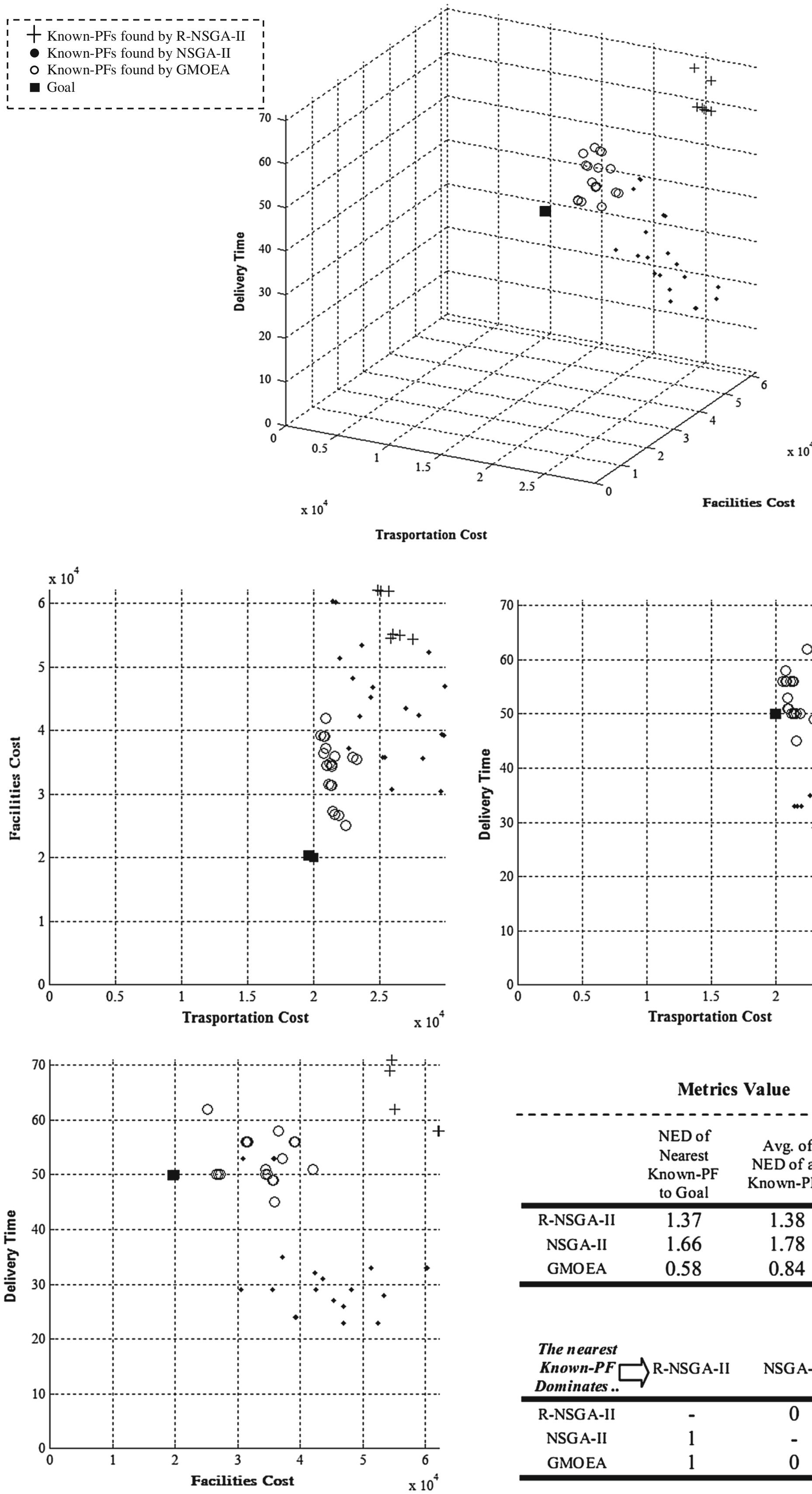
Through the illustrative example, GMOEA found a solution that is closer to the goal that two other solutions found by R-NSGA-II and NSGA-II. In addition, the solution given by GMOEA has objectives value better that or (at least) equal to the corresponding objectives value of the solutions given by R-NSGA-II and NSGA-II.

Since evolutionary algorithms cannot be statistically evaluated through a single run, more runs are needed to evaluate the performance of GMOEA. Therefore, the three algorithms were run to solve the problem more than one time, and the statistical results are presented.

R-NSGA-II, NSGA-II and GMOEA were additionally executed with 100 independent runs on the fLN problem to search for the nearest solutions to the underestimated goal. During 500 generations in each run, the related metrics including NED of the nearest known-PF, the average of NED of all known-PFs, the numbers of known-PFs were recorded. Thereafter, the descriptive results, with 95 % CI of the mean of the recorded values, are presented in Fig. 9 in each 100 generations.

The nearest known-PF (to the goal) found by GMOEA is always closer than its corresponding known-PFs found by other algorithms to the underestimated goal at generations of 100, 200, 300, 400 and 500 (see Fig. 9a).

In addition, all known-PFs found by GMOEA are also closer to the goal, on average, compared to other known-PFs found by R-NSGA-II and NSGA-II (see Fig. 9b).



Metrics Value

	NED of Nearest Known-PF to Goal	Avg. of NED of all Known-PFs	Known-PFs Count
R-NSGA-II	1.37	1.38	7
NSGA-II	1.66	1.78	20
GMOEA	0.58	0.84	20

The nearest Known-PF Dominates ..	R-NSGA-II	NSGA-II	GMOEA
R-NSGA-II	-	0	0
NSGA-II	1	-	0
GMOEA	1	0	-

Fig. 8 Known-PFs found by three algorithms on the fLN Problem and their metrics value with the underestimated goal

Table 1 fLN design problem information

# of suppliers	# of distributing centres	# of retailers	# of customers	Underestimated goal			Overestimated goal		
5	8	10	20	FC*	TC*	T*	FC	TC	T
				20,000	20,000	50	30,000	60,000	50

* *FC* facility cost * *TC* transportation cost * *T* (delay) time

Table 2 Evolutionary computing setting

Population size	Crossover rate	Mutation rate	Crossover type	Mutation type
20	0.95	0.1	Double point crossover	Randomly change the number of one item in string

Table 3 Know-PFs found R-NSGA-II (underestimated goal= [20,000 20,000 50])

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7
Trans. cost	25,934	26,474	25,780	27,470	24,843	25,080	25,620
Fac. cost	55,066	55,006	54,579	54,342	62,137	61,979	61,919
Del. time	62	62	71	69	58	58	58

Considering the size of populations that is fixed to 20, the number of known-PFs at different generations in GMOEA is noticeable more (around 17 out of 20) than the number of known-PFs found by R-NSGA-II (around 3 out of 20); however, it is not more than the corresponding number in NSGA-II (around 19 out of 20) (see Fig. 9c).

The quality of nearest known-PFs to the goal must be relatively evaluated, since their closeness to the goal and their objectives value are simultaneously considered. Based on the expected quality of the nearest known-PFs to the underestimated goal, six different cases were defined for each pair of the solutions (nearest known-PFs). Each case indicates a particular situation that two given solutions can be easily compared to each other. The cases are listed and described in Tables 6.

Each case is given a percentage value meaning that in what percentage of all runs (of 100 runs) the case is occurred. With the cases defined in Table 6, Algorithm A outperforms Algorithms B if there are noticeable higher percentages, first, in Case 1, then, Case 2. B outperforms A if there are noticeable higher percentages, first in Case 6, then, Case 5. Otherwise, both algorithms have no superiority over each other (Case 3 and Case 5).

The percentages of the cases related to the fLN problem, are presented in Fig. 10. GMOEA gives a better solution in 27 % of all runs in which the solution dominates its corresponding solution found by R-NSGA-II, while it is closer to the goal (Case 1 (GMOEA vs. R-NSGA-II)). And in more general state, GMOEA found a better solution in 87 % of all runs in which the solution is closer to the goal compared to its corresponding solution found by R-NSGA-II, while the solution is not dominated, at least, by its corresponding solution given by R-NSGA-II (27 + 60 % = 87 % in Cases 1 and 2 of GMOEA vs. R-NSGA-II). The similar findings are with the comparison of GMOEA and NSGA-II (in Cases 1 and 2, 80 % of runs).

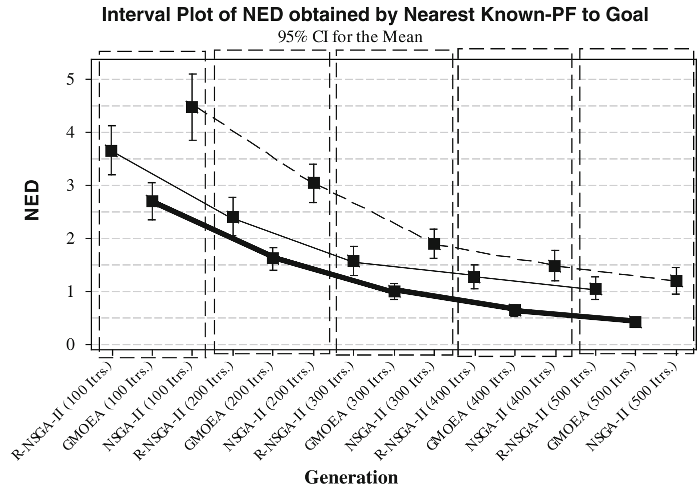
GMOEA outperforms R-NSGA-II and NSGA-II in 87 % and 80 % of all runs, respectively, on the fLN problem with the underestimated goal. In other words, in more that 80 % of the 100 runs (=80 runs), GMOEA is able to found a closer solution to the goal, as its objectives have better value than, or the same value with their corresponding objectives of the nearest solutions found by two other algorithms.

The same efforts and analysis must be, however, taken with the overestimated goal, which is not similar to the aforementioned analysis. It is discussed in the next subsection.

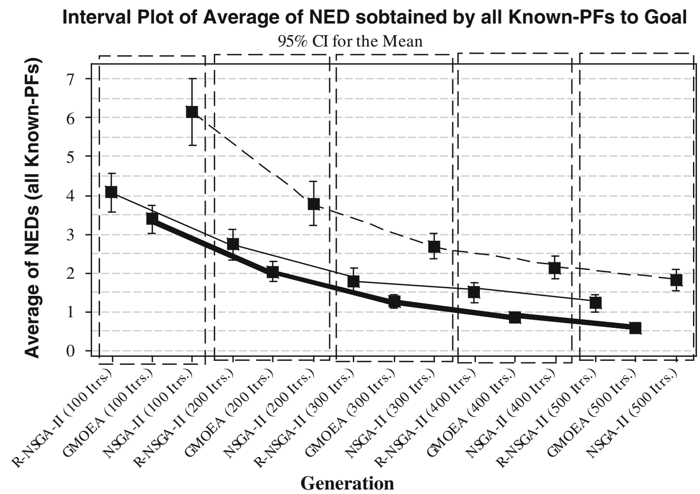
Testing with overestimated goal

In this section, the problem, fLN was considered with an overestimated goal with a desired transportation cost (=30,000), a desired facility cost (=30,000) and a desired maximal delivery time (=50). All three algorithms used in the previous experiment were run to the problem, and their known-PFs found in their populations after 500 iterations, are illustrated in Fig. 11. The NED of the nearest known-PF to the goal, the average of NED of all the known-PFs, the number of these solutions and the quality of the three nearest known-PFs (in

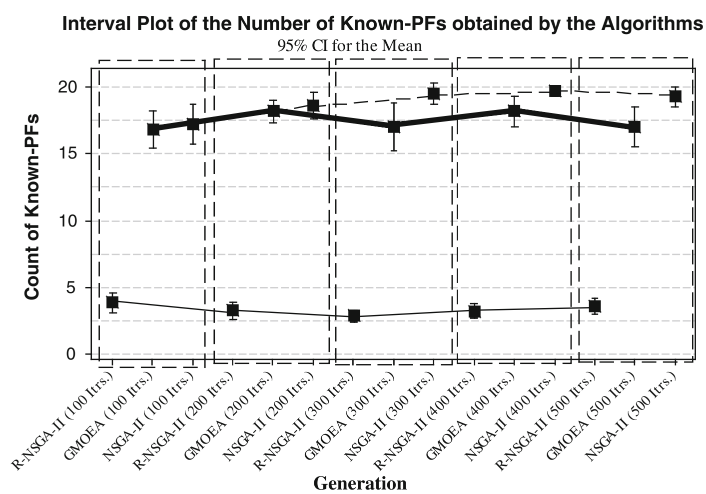
Fig. 9 Known-PFs found by three algorithms on the fLN Problem with the underestimated goal



(a) NED of nearest known-PF (to goal) found by three algorithms at different generations



(b) Average of NED of known-PFs found by three algorithms at different generations



(c) Number of known-PFs found by three algorithms at different generations

Table 4 Known-PF found by NSGA-II

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	28,661	29,693	28,218	29,558	25,902	24,406	21,378	21,630	21926	22,610
Fac. cost	52,333	39,341	35,595	30,476	30,790	46,862	60,294	60,120	51,296	37,202
Del. time	23	24	29	29	53	26	33	33	33	35
	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF 17	K-PF 18	K-PF 19	K-PF 20
Trans. cost	29,884	25,232	26,910	22,933	23,577	29,789	24,311	27,916	23,443	25,328
Fac. cost	46,886	35,842	43,552	48,198	53,401	39,245	45,280	42,442	42284	35746
Del. time	23	53	31	29	28	24	27	29	32	53

Table 5 Known-PFs found by GMOEA (underestimated goal= [20,000 20,000 50])

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	22,411	21,265	21,529	21,347	21,395	21,343	21,307	21,213	21,079	21,853
Fac. cost	25,152	31,435	26,885	34,405	27,287	31,383	34,763	34,807	31,559	26,642
Del. time	62	56	50	50	50	56	50	50	56	50
	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF 17	K-PF 18	K-PF 19	K-PF 20
Trans. cost	22,920	23,244	20,972	20,786	20,708	20,522	20,884	21,536	20,708	20,884
Fac. cost	35,726	35,483	34,523	39,021	39,073	39,197	41,999	35,876	36,467	37,201
Del. time	49	49	51	56	56	56	51	45	58	53

Table 6 Six cases of comparing two nearest known-PFs given by two algorithms (A and B) with an underestimated goal

Case	Description
Case 1 (A vs. B)	The solution given by A **dominates** and**is closer** than the solution given by B
Case 2 (A vs. B)	The solution given by A is **non-dominated** and**closer** than the solution given by B
Case 3 (A vs. B)	The solution given by A is **dominated** by and**closer** than the solution given by B
Case 4 (A vs. B)	The solution given by A **dominates** and**is not closer** than the solution given by B
Case 5 (A vs. B)	The solution given by A is **non-dominated** and**not closer** than the solution given by B
Case 6 (A vs. B)	The solution given by A is **dominated** by and**not closer** than the solution given by B

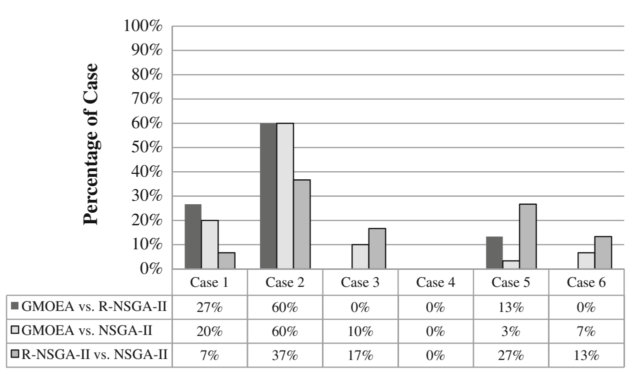
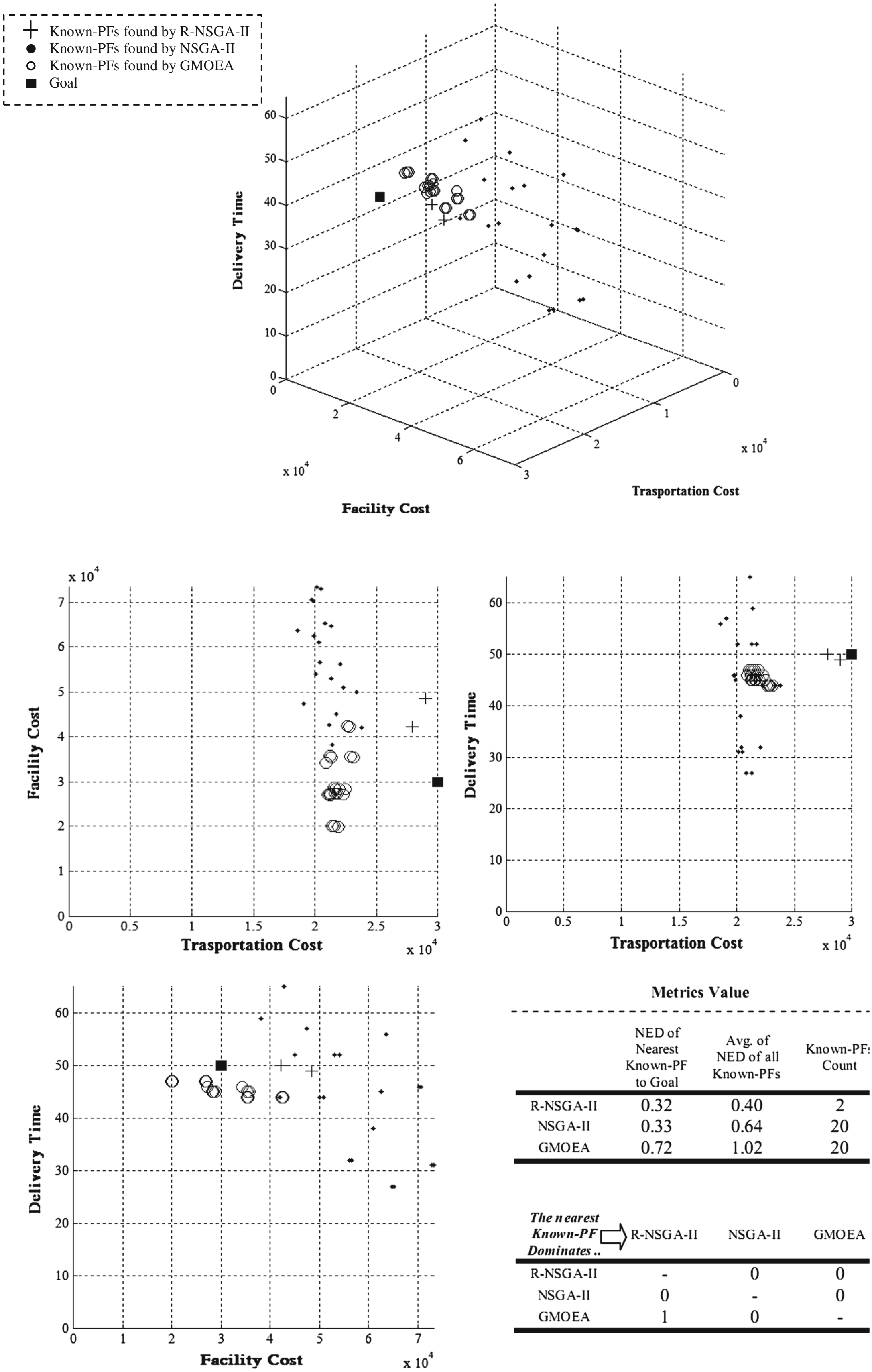


Fig. 10 The round percentage of the six cases of comparison each two nearest known-PFs given by the algorithms on the fLN Problem with the underestimated goal

terms on their objective values) are presented in the metric data section in the figure.

GMOEA found a nearest known-PF with the NED of 0.72, and it is not close to the goal compared to its corresponding solutions found by R-NSGA-II and NSGA-II (NED in R-NSGA-II = 0.32 and in R-NSGA-II = 0.33). The three set of known-PFs found by the algorithms are presented in Tables 7, 8 and 9, and the nearest known-PF given by each algorithm is marked (in gray color) in its related table. Although GMOEA found a solution not close to the goal compared to the solution given solution by R-NSGA-II, its solution has better objectives value (transportation cost = 23,102, facility cost = 35,395 and maximal delivery time=44) as it dominates the solution given by R-NSGA-II (transportation cost = 27,941, facility cost = 42,222 and maximal delivery time=50).

The closeness is not important in this situation, since no decision-maker prefers a solution (even close to the goal) to another solution that has better objective values (*better objec-*



Metrics Value

	NED of Nearest Known-PF to Goal	Avg. of NED of all Known-PFs	Known-PFs Count
R-NSGA-II	0.32	0.40	2
NSGA-II	0.33	0.64	20
GMOEA	0.72	1.02	20

The nearest Known-PF Dominates ..	R-NSGA-II	NSGA-II	GMOEA
R-NSGA-II	-	0	0
NSGA-II	0	-	0
GMOEA	1	0	-

Fig. 11 Known-PFs found by three algorithms on the fLN problem and their metrics value with the overestimated goal

tive values: less value in at least one objective than, and/or the same value in its other objectives with the corresponding objectives in the closer solution).

The decision-maker might pessimistically specify the goal at initial stag of optimization, and now, the proposed guided algorithm gives him or her, at least, a better view of the objective space that has much better solutions, and even close to the interested values. Therefore, NED cannot be solely considered to evaluate the performance of the guided algorithms in this situation.

R-NSGA-II, NSGA-II and GMOEA were executed with 100 independent runs for the fLN problem considering the overestimated goal. The NED of the nearest known-PFs, the average of NED of all known-PFs and the number of these known-PFs were recorded at each 100 generations. The descriptive result with 95 % CI of the recorded values mean is presented in Fig. 12.

The nearest known-PF found by GMOEA is not closer to the goal than its corresponding solution found by R-NSGA-II and NSGA-II at different generations (see Fig. 12a). Because GMOEA first tries to search for optimal solutions, and then, their closeness to the goal is taken to account based on the guiding strategy considered in GMOEA. Since the goal has overvalued objectives, many feasible solutions including optimal ones can sdominate the goal. Therefore, the NED values shown in Fig. 12a cannot show the algorithm performance alone in all situations. In other words, a good solution, here, must be first optimal (or a solution that is close to an optimal solution as possible), and then, close to the goal as can as possible.

There are similar finding with all known-PFs found by the algorithms (see Fig. 12b). The average of NED is increased as the generation continuous; and it means the algorithms found the solutions, which are located at the distance (form the goal). In additions, the number of known-PFs found by each algorithm at different generations is presented in Fig. 12c. The number of known-PF found by GMOEA and NSGA-II are noticeable more than the corresponding number in R-NSGA-II (around three known-PFs) at different generations.

Since the quality of nearest known-PFs cannot be specified based on NED metric alone, the quality must be indicated with their objectives value and closeness to the goal simultaneously. Searching for such these solutions is another optimization effort with two conflicting objectives. Although, GMOEA is designed to deal with any type of goal, whether is under estimated or overestimated, the evaluation and comparison should be different from the previous experiment.

Four cases, similar to the previous experiment, were considered; however, two cases must be integrated into other two cases. Because the closeness of two given nearest known-PFs to the goal is not important, while one solution dominates another one. The dominated solution is always considered as a worst solution. The four cases are described in Table 10. Each Case is assigned to a percentage value meaning what percentage of all runs the case is occurred.

The percentages of all cases gained through running 100 independent runs of the three algorithms on the fLN problem are shown in Fig. 13. According to the two first cases, Cases 1 and 2, in 60 % of all runs (60 out of 100 runs), GMOE found a better solution compared to R-NSGA-II. The solution either dominates (Case 1) or is closer to the goal than the nearest known-PFs given by R-NSGA-II while the solutions do not dominate each other (Case 2). Meanwhile, R-NSGA-II found a solution with the same situation only in 40 % of all runs (Cases 3 and 4).

GMOEA also found a better solution in 63 % (43 % in Case 1 + 20 % in Case 2) of all runs compared to NSGA-II. The solution dominates the known-PF found by NSGA-II as a nearest solution to the goal, or it is closer to the goal while both solutions are not dominated by each other.

In addition, there is no significant difference, in general, between the quality of two solutions as two nearest known-PFs to the goal given by R-NSGA-II and NSGA-II. R-NSGA-II outperforms NSGA-II in 54 % of runs (Cases 1 and 2), while the percentage is 46 % for NSGA-II.

Base on the above findings, the proposed guided MOEA outperforms an existing guided MOEA, R-NSGA-II and a non-guided MOEA, NSGA-II. However, the result shows no substantial advantage with GMOEA in which the algorithm has to deal with overestimated goals compared to the two other algorithms.

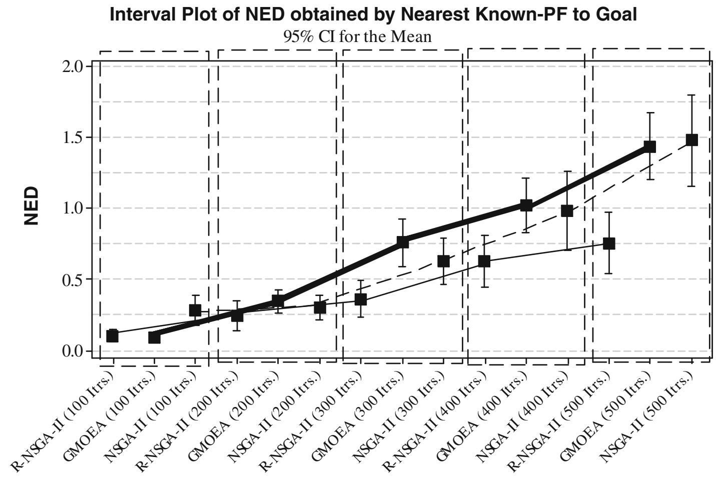
In the two next subsections, the two extreme goals were additionally considered to evaluate the performance of GMOEA comparing to R-NSGA-II and NSGA-II. Therefore, the goals with the values of possibly minimal values for all objectives (transportation cost=1, facility cost=1 and maximal delivery time=1) and with the maximum values (are equal to 100,000) for all objectives (transportation cost=100,000, facility cost=100,000 and maximal delivery time=100,000) are considered as extremely underestimated and overestimated goals, respectively.

Testing with extremely underestimated goal

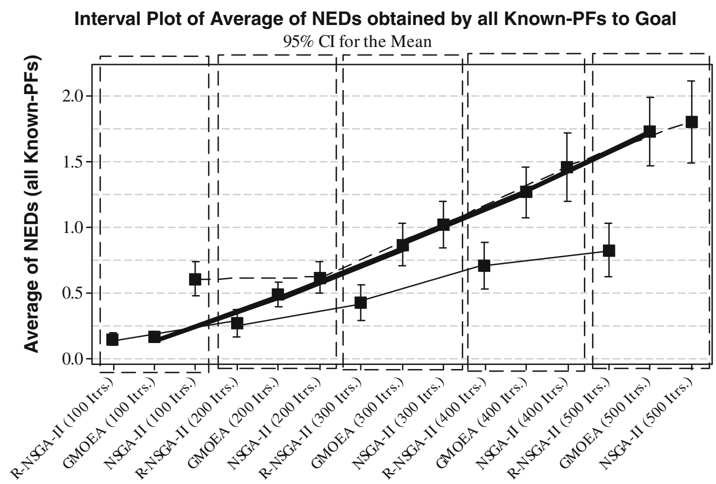
In this section, GMOEA, R-NSGA-II and NSGA-II were use to solve the fLN problem with the extremely underestimated goal of a desired transportation cost (=1), a desired facility cost (=1) and a desired maximal delivery time (=1). Similar to the previous subsections, known-PF found by the three algorithms after 500 generations are shown in Fig. 14 and Tables 11, 12 and 13.

GMOEA still shows better performance in this illustrative example, since it found closer known-PF to the extremely

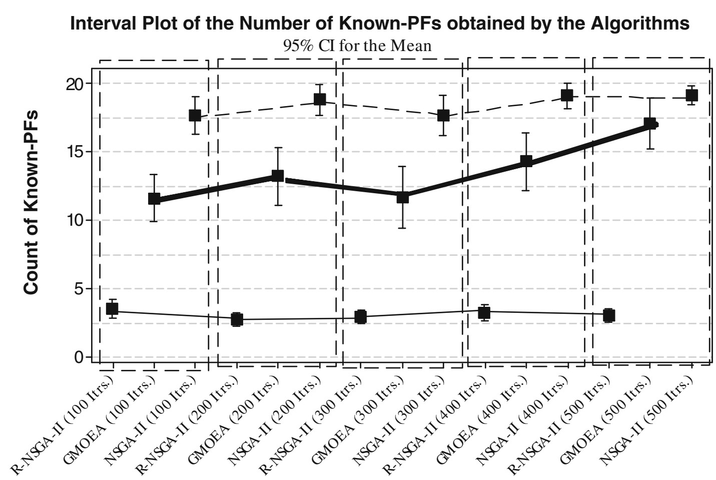
Fig. 12 Known-PFs found by three algorithms on the fLN Problem with the overestimated goal



(a) NED of nearest known-PF (to goal) found by three algorithms at different generations



(b) Average of NED of known-PFs found by three algorithms at different generations



(c) Number of known-PFs found by three algorithms at different generations

Table 7 Know-PFs found R-NSGA-II (Goal = [30,000 30,000 50])

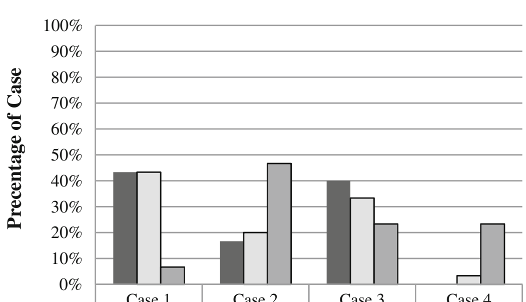
	K-PF 1	K-PF 2
Trans. cost	29,026	27,941
Fac. cost	48,435	42,222
Del. time	49	50

Table 8 Known-PF found by NSGA-II

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	18,617	20,186	21,307	19,093	20,390	22,300	20,302	23,817	21,758	19,819
Fac. cost	63,588	73,242	64,688	47,390	56,500	50,884	60,962	42,041	45,046	70,235
Del. time	56	31	27	57	32	44	38	44	52	46
	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF 17	K-PF 18	K-PF 19	K-PF 20
Trans. cost	19,749	23,377	20,824	21,318	21,443	21,179	22,065	20,472	19,921	20,083
Fac. cost	70,581	50,010	65,189	53,015	38,182	42,732	56,175	72,894	62,516	54,047
Del. time	46	44	27	52	59	65	32	31	45	52

Table 9 Known-PFs found by GMOEA (Goal= [30,000 30,000 50])

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	21,561	22,001	21,570	21,649	21,227	21,922	22,926	21,394	21,843	21,251
Fac. cost	28,798	28,358	19,981	28,446	27,341	19,893	35,571	20,157	27,253	35,721
Del. time	45	45	47	45	46	47	44	47	46	45
	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF 17	K-PF 18	K-PF 19	K-PF 20
Trans. cost	21,339	22,445	21,260	21,084	20,917	21,671	22,287	22,616	23,102	22,792
Fac. cost	35,369	28,284	26,904	27,080	34,264	27,267	27,179	42,494	35,395	42,318
Del. time	45	45	47	47	46	46	46	44	44	44



	Case 1	Case 2	Case 3	Case 4
GMOEA vs. R-NSGA-II	43%	17%	40%	0%
GMOEA vs. NSGA-II	43%	20%	33%	3%
R-NSGA-II vs. NSGA-II	7%	47%	23%	23%

Fig. 13 The round percentage of the four cases of comparison each two nearest known-PFs given by the algorithms on the fLN Problem with the overestimated goal

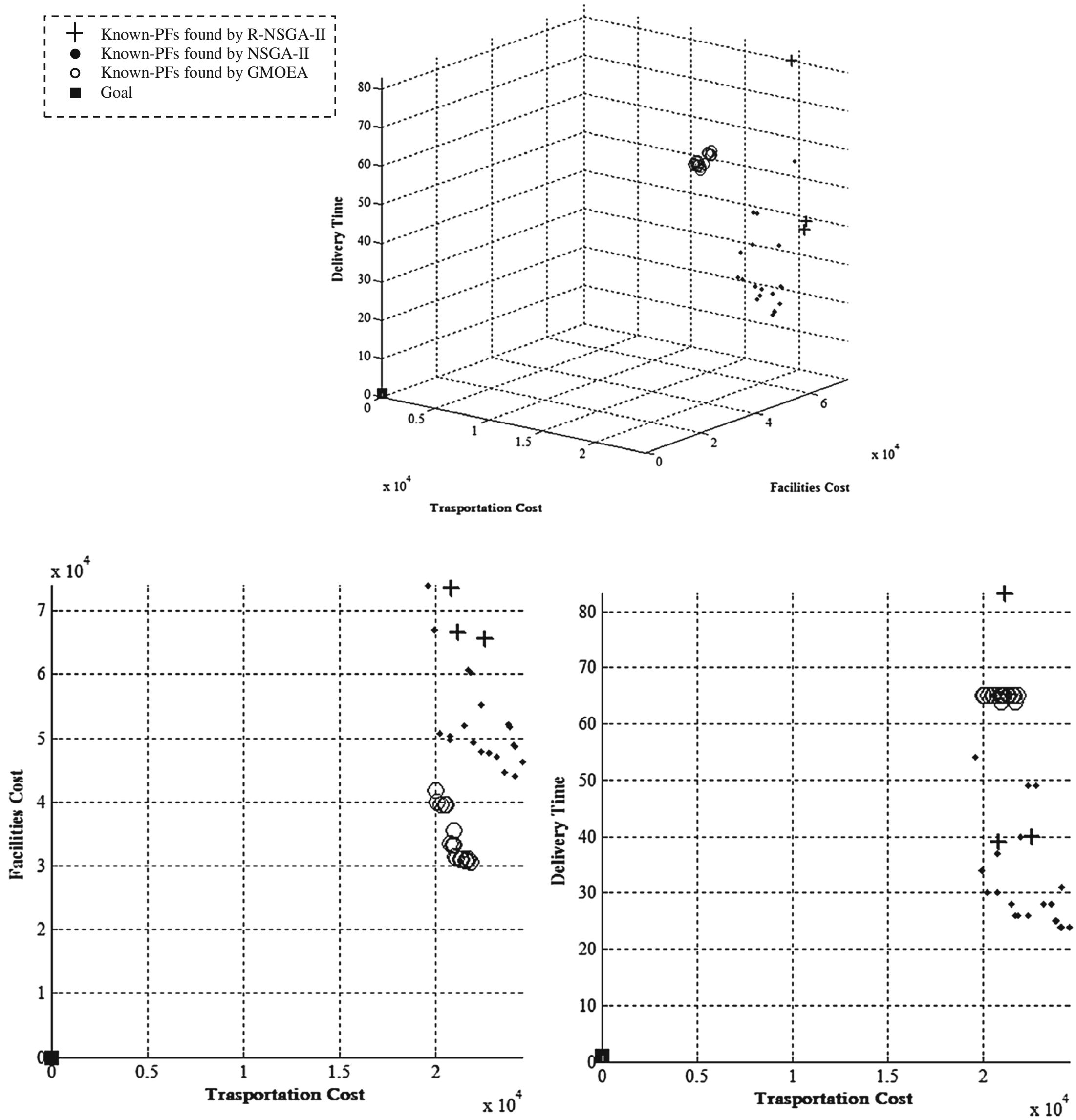
underestimated goal with the NED of 35.18 comparing to other algorithms (45.04 in R-NSGA-II and 41.44 in NSGA-II). The solution found by GMOEA is also not dominated by two other solutions found by R-NSGA-II and NSGA-II.

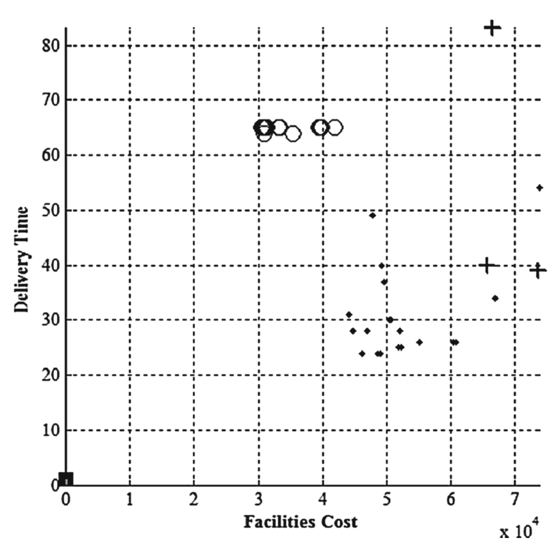
The average of NED of known-PFs given by GMOEA is 38.64, while this value is 44.85 in R-NSGA-II and 49.86 in NSGA-II. All known-PFs found by GMOEA are also closer to the goal, on average, compared to other sets of known-PFs found by two other algorithms.

To evaluate GMOEA statistically, all three algorithms were also executed on the problem with 100 independent runs, and the related results are presented in Figs. 15 and 16.

The guided MOEAs, GMOEA and R-NSGA-II found closer solutions to the goal at early generation. Since the generation continues, all algorithms found the solutions with nearly the same distance from the goal (see Fig. 15a). However, GMOEA has better performance compared to NSGA-II and R-NSGA-II in terms of the average of NED and the number of known-PFs in population, respectively goal (see Fig. 15b, c).

All the aforementioned six cases (described early) were considered with the new extremely underestimated goal. The percentages for all cases are shown in Fig. 16.





Metrics Value

	NED of Nearest Known-PF to Goal	Avg. of NED of all Known-PFs	Known-PFs Count
R-NSGA-II	45.04	44.85	3
NSGA-II	41.44	49.86	20
GMOEA	35.18	38.64	17

The nearest Known-PF Dominates ..	R-NSGA-II	NSGA-II	GMOEA
R-NSGA-II	-	0	0
NSGA-II	1	-	0
GMOEA	0	0	-

Fig. 14 Known-PFs found by three algorithms on the fLN Problem and their metrics value with the extremely underestimated goal

Table 10 Four cases of comparing two nearest known-PFs given by two algorithms (A and B) with an overestimated goal

Case Name	Description
Case 1 (A vs. B)	The solution given by A **dominates** the solution given by B
Case 2 (A vs. B)	The solution given by A is **non-dominated** and **closer** than the solution given by B
Case 3 (A vs. B)	The solution given by A is **non-dominated** and **not closer** than the solution given by B
Case 4 (A vs. B)	The solution given by A is **dominated** by the solution given by B

Table 11 Know-PFs found R-NSGA-II (underestimated goal= [1 1 1])

	K-PF 1	K-PF 2	K-PF 3
Trans. cost	20,827	21,197	22,607
Fac. cost	73,547	66,495	65,646
Del. time	39	83	40

Table 12 Known-PF found by NSGA-II

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	24,093	23,883	24,565	23,225	24,166	21,688	21,502	20,262	19,640	20,771
Fac. cost	48,972	51,820	46,253	47,015	44,095	60,705	51,963	50,723	73,819	50,409
Del. time	24	25	24	28	31	26	28	30	54	30
	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF 17	K-PF 18	K-PF 19	K-PF 20
Trans. cost	19,952	22,369	22809	24181	23,795	22,370	23,609	20,771	22,016	21,845
Fac. cost	66,949	47,881	47,705	48,620	52,172	55,138	44,648	49,716	49,218	60,353
Del. time	34	49	49	24	25	26	28	37	40	26

According to two percentages with Case 1 and Case 2, GMOEA found a closer solution to the goal in half of all runs (10 % in Case 1 + 40 % in Case 2), while the solution is not, at least, dominated by the nearest known-PF given by R-NSGA-II. in addition, GMOEA and NSGA-II found own nearest solutions close to the goal with similar situation. In nearly 53 % of all runs, GMOEA found better solutions, as NSGA-II found better solutions in nearly 47 % of all runs. Meanwhile, NSGA-II found better solution in more than 63 % of all runs compared to R-NSGA-II gave better solution in nearly 37 % of all runs.

Based on the percentages indicated by the cases, the proposed guided MOEA, GMOEA has no substantial advantage when it has to deal with an extremely underestimated goal compared to R-NSGA-II and NSGA-II. However, it found more known-PFs around the goal that are also closer to the goal on average. GMOEA is able to search for not only a nearest optimal solution to the goal, but also more optimal solutions near to the goal in the fLN problem. It might help respective decision-maker to have more optimal options to select.

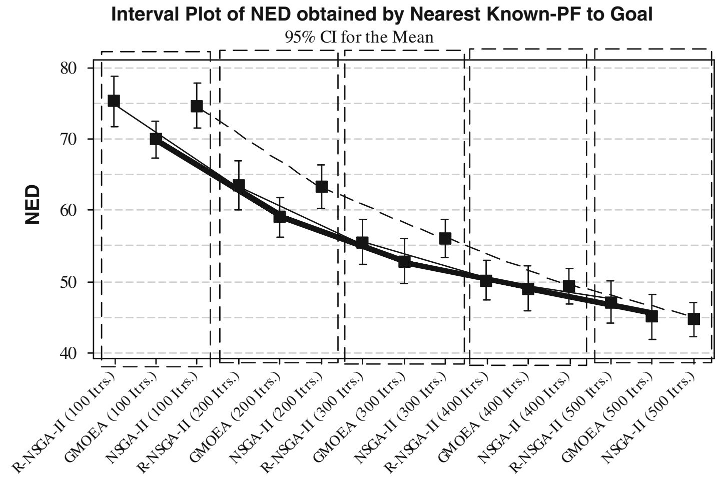
Testing with an extremely overestimated goal

In this section, GMOEA, R-NSGA-II and NSGA-II were use to solve the fLN problem with a different goal that is extremely overestimated in transportation cost (=100,000), facility cost (=100,000) and maximal delivery time (=100,000). Similar to the previous subsections, known-PF found by the three algorithms after 500 generations are shown in Fig. 17 and Tables 14, 15 and 16.

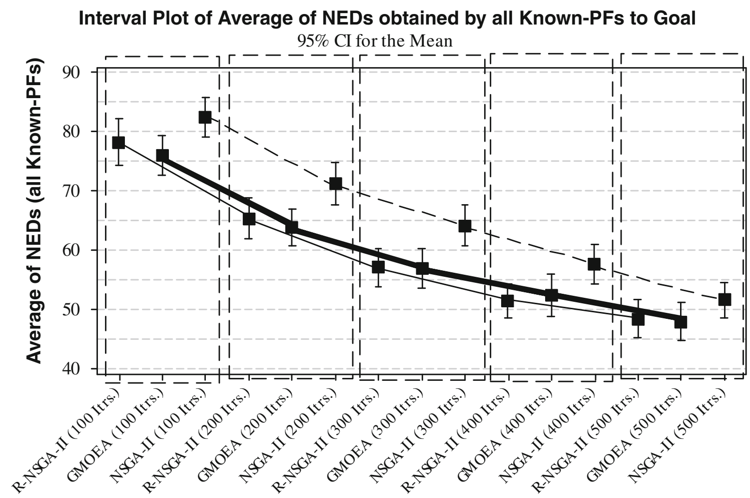
GMOEA found a closer solution to the extremely overestimated goal with the NED of 3,299,791 comparing to other nearest solutions given by R-NSGA-II and NSGA-II (3,300,181 in R-NSGA-II and 3,300,388 in NSGA-II). Since the nearest solutions found by the three algorithms are not dominated by each other (see the metric value section in Fig. 17), the solution given by GMOEA is preferred based on its NED.

Additionally, the average of NED of all known-PFs given by GMOEA is 3,299,819, while the corresponding value is 3,300,183 and 3,302,615 for R-NSGA-II and NSGA-II, respectively. It shows that all known-PFs found by GMOEA

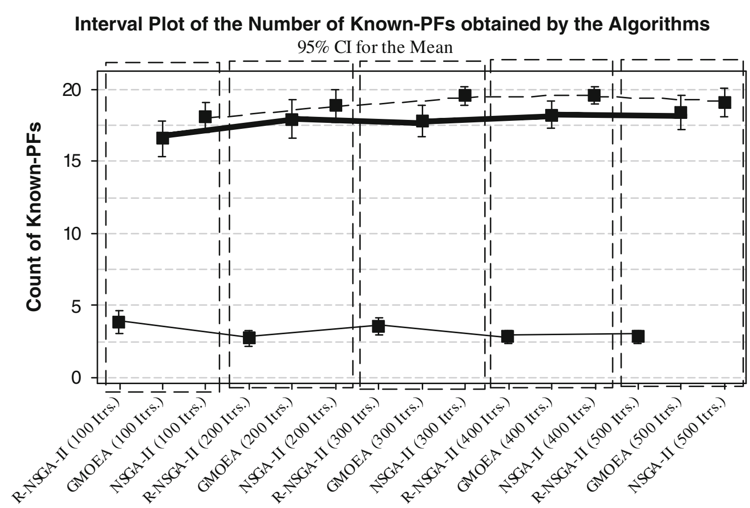
Fig. 15 Known-PFs found by three algorithms on the fLN Problem with the extremely underestimated goal



(a) NED of nearest known-PF (to goal) found by three algorithms at different generations



(b) Average of NED of known-PFs found by three algorithms at different generations



(c) Number of known-PFs found by three algorithms at different generations

Table 13 Known-PFs found by GMOEA (underestimated goal= [1 1 1])

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	20,331	20,040	20,595	20,493	20,907	20,745	20,130	21,559	21,009	21,133
Fac. cost	39,651	41,855	39,475	39,489	33,259	33,421	39,919	30,782	33,245	31,120
Del. Time	65	65	65	65	65	65	65	65	65	65

	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF 17
Trans. cost	21,295	21,823	21,397	21,661	21,066	21,009	21,745
Fac. cost	30,958	30,606	30,944	30,768	31,321	35,385	30,968
Del. time	65	65	65	65	65	64	64

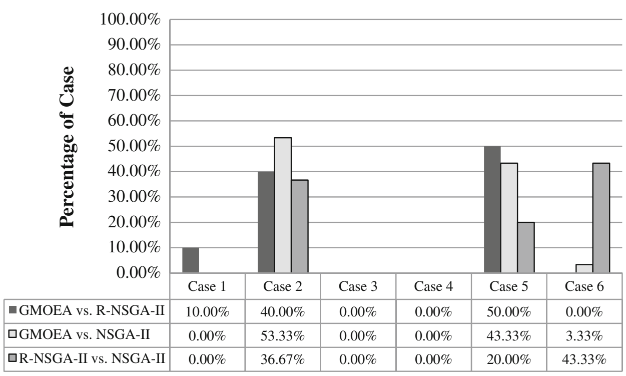


Fig. 16 The round percentage of the four cases of comparing each two nearest known-PFs given by the algorithms on the fLN Problem with the extremely underestimated goal

are also closer to the goal compared to two other sets of known-PFs (see Fig. 17).

GMOEA and NSGA-II additionally give more options to respective decision-makers, since they found 20 known-PFs (= population size) in its final populations; whereas, R-NSGA-II found only three known-PFs in its final population (see Fig. 17).

In order to perform statistical analysis, all three algorithms were additionally executed on the problem with 100 independent runs, and the related results are presented in Figs. 18 and 19.

R-NSGA-II not only found a closer solution to the goal (Fig. 18a), but also it reached less value in the NED average (of its known-PFs) at all generations (Fig. 18b) compared to two other algorithms. However, the related diagram does not show the exact quality of the algorithm and its given solutions. R-NSGA-II found substantially fewer numbers of known-PFs (nearly three) during generation than GMOEA and NSGA-II (between 16 and 20) (see Fig. 18c).

The algorithms were also compared based on the four cases (described early) through 100 individual runs. The related percentages are presented in Fig. 19. Interestingly, GMOEA is still better than R-NSGA-II in 40 % of all runs in reaching a closer solution with the ability of dominating the nearest known-PF found by R-NSGA-II (Case 1 of GMOEA vs. R-NSGA-II). In addition, in 20 % of all runs, both algorithms found the solutions that are not dominated by each other; however, the solution given by GMOEA is closer to the goal. Thus, in 60 % of runs, GMOEA shows better performance when it deals with the extremely overestimated goal.

Furthermore, the proposed guided MOEA, GMOEA found a closer solution (to the goal) than NSGA-II in 90 % of all runs, while two given nearest solutions by both algorithms cannot be compared based on their objectives value (Case 2 of GMOEA vs. NSGA-II).

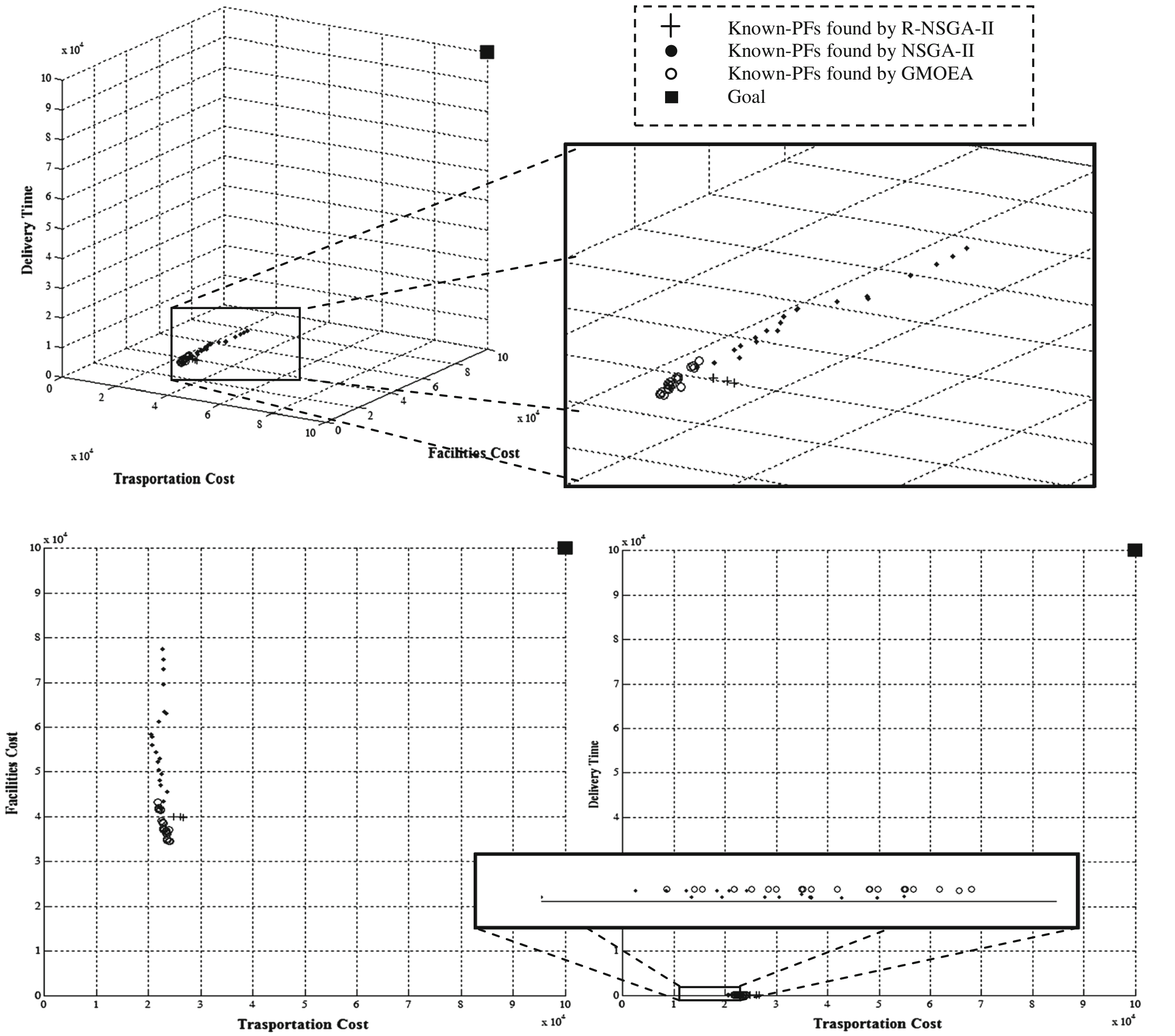
R-NSGA-II found a better solution in 77 % of all runs (Case 2 of R-NSGA-II vs. NSGA-II) than NSGA-II (with only 23 % of all runs, the related Case 3 + Case 4).

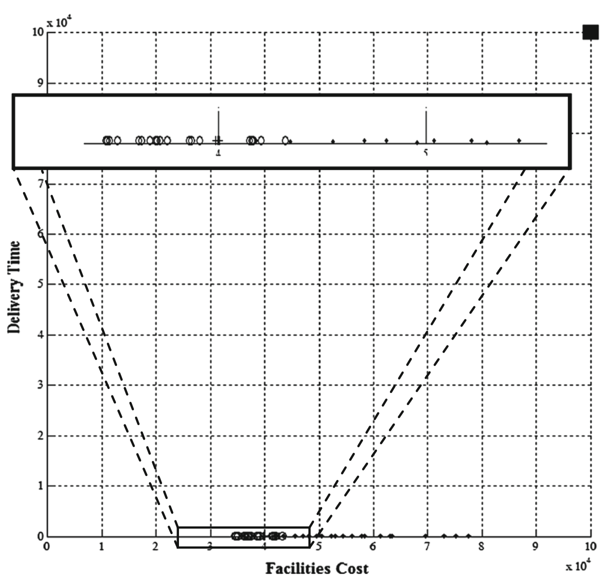
According to the recent finding and the previous finding, GMOEA shows better performance in the most runs with the overestimated goal, whether is extreme or not, compared to R-NSGA-II and NSGA-II.

Discussion and conclusion

In this research, the guided MOEA, GMOEA was proposed and tested through solving a presented multi-objective optimization problem of flexible logistics network design. Four different goals have been considered for the problem. Known-PFs that were found by GMOEA were compared to other known-PFs found by two other well-known guided and non-guided MOEAs, R-NSGA-II and NSGA-II at different numbers of generation.

Based on the experimental results, GMOEA is able to find a set of known-PFs closer (on average) to an underestimated goal in the fLN problem, while the number of these solutions is nearly equal to its population size. In additions, GMOEA has 87 chances in 100 (with probability of 0.87) to find a closer solution to the underestimated goal with better or equal objective values compared to the solution found by R-NSGA-II. However, the difference between the closeness of two





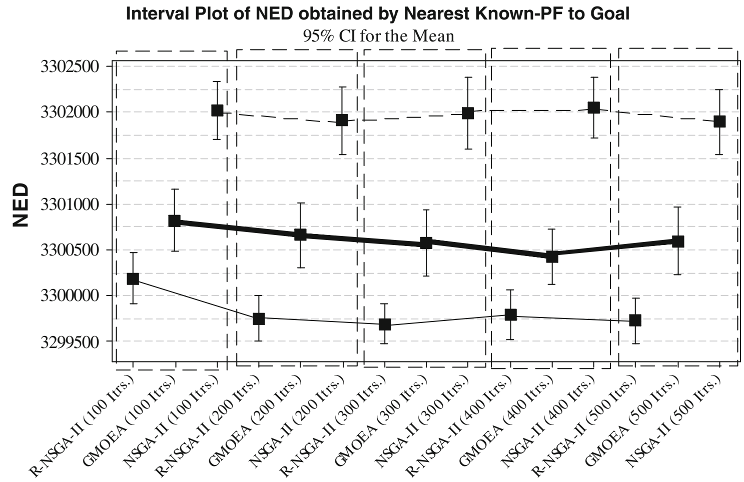
Metrics Value

	NED of Nearest Known-PF to Goal	Avg. of NED of all Known-PFs	Known-PFs Coun
R-NSGA-II	3,300,181	3,300,183	3
NSGA-II	3,300,388	3,302,615	20
GMOEA	3,299,791	3,299,819	20

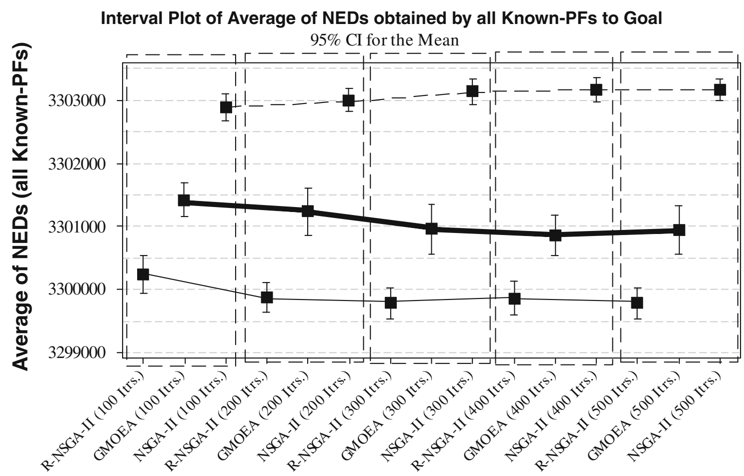
The nearest Known-PF Dominates ..	R-NSGA-II	NSGA-II	GMOEA
R-NSGA-II	-	0	0
NSGA-II	0	-	0
GMOEA	0	0	-

Fig. 17 Known-PFs found by three algorithms on the fLN Problem and their metrics value with the extremely overestimated goal

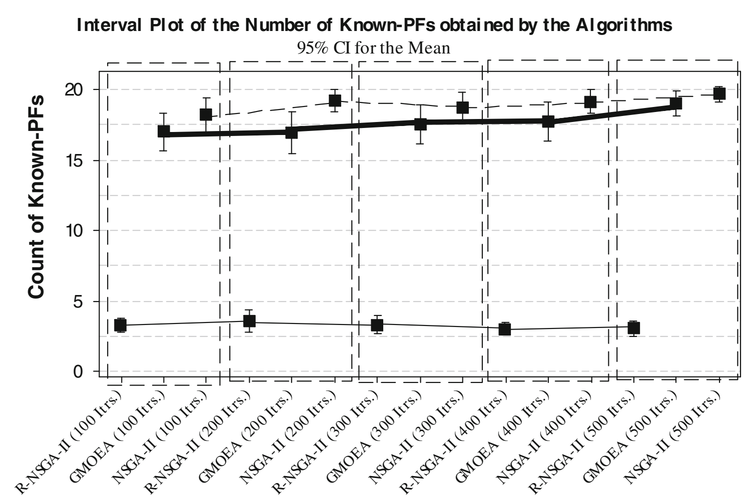
Fig. 18 Known-PFs found by three algorithms on the fLN Problem with the extremely overestimated goal



(a) NED of nearest known-PF (to goal) found by three algorithms at different generations



(b) Average of NED of known-PFs found by three algorithms at different generations



(c) Number of known-PFs found by three algorithms at different generations

Table 14 Know-PFs found R-NSGA-II (Goal= [100,000 100,000 100,000])

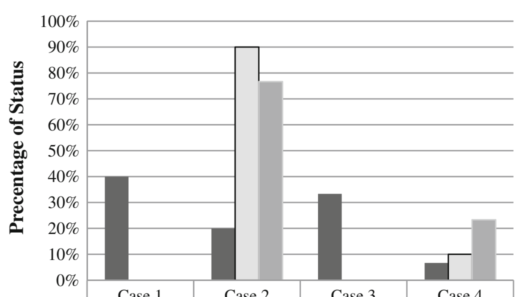
	K-PF 1	K-PF 2	K-PF 3
Trans. cost	26,658	26,010	24,814
Fac. cost	39,884	39,965	40,057
Del. time	86	86	86

Table 15 Known-PF found by NSGA-II

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	24,088	23,840	23,564	23,994	23,577	21,996	22,380	23,639	22,507	22,783
Fac. cost	34,558	34,620	36,171	37,035	35,126	41,673	41,481	34,754	39,090	37,539
Del. time	92	92	92	84	92	92	92	92	92	92
	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF 17	K-PF 18	K-PF 19	K-PF 20
Trans. cost	22,244	23,046	23,294	23,363	21,934	23,301	22,770	22,845	21,720	22,569
Fac. cost	41,611	37,033	36,971	36,305	42,045	36,677	38,584	37,167	43,224	38,718
Del. time	92	92	92	92	92	92	92	92	92	92

Table 16 Known-PFs found by GMOEA (Goal= [100,000 100,000 100,000])

	K-PF 1	K-PF 2	K-PF 3	K-PF 4	K-PF 5	K-PF 6	K-PF 7	K-PF 8	K-PF 9	K-PF 10
Trans. cost	23,353	22,844	22,844	23,081	20,508	20,745	22,771	22,596	22,833	21,909
Fac. cost	63,140	69,595	73,101	63,448	58,308	56,015	43,465	77,512	75,219	61,258
Del. time	27	29	28	27	33	33	54	30	30	32
	K-PF 11	K-PF 12	K-PF 13	K-PF 14	K-PF 15	K-PF 16	K-PF17	K-PF 18	K-PF 19	K-PF 20
Trans. cost	22,342	23,562	22,146	20,702	22,481	21,872	21,715	22,207	22,109	21,478
Fac. cost	41,829	45,533	52,939	57,920	49,582	50,414	52,207	47,057	48,121	54,500
Del. time	83	39	32	62	32	83	83	83	83	83



	Case 1	Case 2	Case 3	Case 4
GMOEA vs. R-NSGA-II	40%	20%	33%	7%
GMOEA vs. NSGA-II	0%	90%	0%	10%
R-NSGA-II vs. NSGA-II	0%	77%	0%	23%

Fig. 19 The round percentage of the four cases of comparison each two nearest known-PFs given by the algorithms on the fLN Problem with the extremely overestimated goal

nearest known-PFs found by GMOEA and R-NSGA-II is decreased, since the goal is extremely underestimated.

The similar advantage is with GMOEA when is compared to NSGA-II. NSGA-II is a non-guided MOEA and is not supposed to find more solutions around a particular point; however, the algorithm was used in all comparisons showing how good the proposed guided MOEA finds solutions around the goal. Based on the findings in this chapter, GMOEA outperforms NSGA-II in finding closer and better solutions (in terms of objectives value) for the fLN problem considering underestimated goals. GMOEA has 80 chance in 100 (with probability of 0.8) to find better solutions compared to NSGA-II in this situation.

GMOEA has also advantage in finding better solutions for the fLN problem with overestimated goals. The algorithm has 60 and 63 chances in 100 to find a better solution compare to R-NSGA-II and NSGA-II, respectively.

The flexible logistics networks have been enhanced through considering the flexibility of shipment over the whole networks. With this flexibility, there are many optimal configurations with the variety of optimal costs and time.

Nonetheless, only a few of these optimal configurations can be practically selected and considered in the network design. Mostly, these are some budget limitations in transportation and facility along with a delivery time that must be considered in the network configuration by logistics providers, particularly third-party logistics providers (3LPs). GMOEA is able to find these solutions, or at least inform respective decision-makers through finding better solutions when the given goal is so over or underestimated.

Acknowledgments The authors would like to thank the Universiti Teknologi Malaysia (UTM), for supporting this research.

References

Branke, J., Kaubler, T., & Schmeck, H. (2000). *Guiding multi-objective evolutionary algorithms toward interesting regions*. Germany: Institute AIFB, University of Karlsruhe.

Branke, J., Kaussler, T., & Schmeck, H. (2001). Guidance in evolutionary multi-objective optimization. *Advances in Engineering Software*, *32*(6), 499–507.

Cagnina, L. C., Esquivel, S. C., & Coello, C. A. C. (2011). Solving constrained optimization problems with a hybrid particle swarm optimization algorithm. *Engineering Optimization*, *43*(8), 843–866.

Cooper, L. (1963). Location-allocation Problems. *Operations Research*, *11*, 331–343.

Cvetkovic, D., & Parmee, I. (2002). Agent-based support within an interactive evolutionary design system. *Ai Edam-Artificial Intelligence for Engineering Design Analysis and Manufacturing*, *16*(5), 331–342.

Cvetkovic, D., & Parmee, I. C. (2002). Preferences and their application in evolutionary multiobjective optimization. *Ieee Transactions on Evolutionary Computation*, *6*(1), 42–57. doi:PiiS1089-778x(02)02206-3.

da Silva, E. K., Barbosa, H. J. C., & Lemonge, A. C. C. (2011). An adaptive constraint handling technique for differential evolution with dynamic use of variants in engineering optimization. *Optimization and Engineering*, *12*(1–2), 31–54.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. West Sussex: Wiley.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *Ieee Transactions on Evolutionary Computation*, *6*(2), 182–197. doi:PiiS1089-778x(02)04101-2.

Deb, K., Sundar, J., Udaya Bhaskara Rao, N., & Chaudhuri, S. (2006). Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms. *International Journal of Computational Intelligence Research*, *2*(3), 273–286. http://repository.ias.ac.in/81053/.

Engelbrecht, A. P. (2007). *Computational intelligence. An introduction* (2nd ed.). London: Wiley.

Fonseca, C., & Fleming, P. (1993). Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In S. Forrest (Ed.), the fifth international conference on genetic algorithms, San Mateo, California, (pp. 416–423). University of Illinois at Urbana Champaign, Morgan Kaufmann Publishers.

Fonseca, C. M., & Fleming, P. J. (1998). Multiobjective optimization and multiple constraint handling with evolutionary algorithms—Part I: A unified formulation. *Ieee Transactions on Systems Man and Cybernetics Part a-Systems and Humans*, *28*(1), 26–37.

Gen, M., Altiparmak, F., & Lin, L. (2006). A genetic algorithm for two-stage transportation problem using priority-based encoding. *Or Spectrum*, *28*(3), 337–354.

Gen, M., Cheng, R., & Lin, L. (2008). *Network models and optimization, multiobjective genetic algorithm approach*. London: Springer.

Gong, M., Liu, F., Zhang, W., Jiao, L., & Zhang, Q. Interactive MOEA/D for multi-objective decision making. In GECCO '11 Proceedings of the 13th annual conference on Genetic and evolutionary computation, (pp. 721–728). New York, NY, USA: ACM. doi:10.1145/2001576.2001675.

Hitchcock, F. L. (1941). The distribution of a product from several sources to numerous localities. *Journal of Mathematical Physics*, *20*, 220–230.

Jin, Y. C., & Sendhoff, B. (2002). Incorporation of fuzzy preferences into evolutionay multiobjective optimization. In The 4th Asia-Pacific conference on simulated evolution and learning, Orchid Country Club, Singapore (vol. 1, pp. 26–30).

Kim, J. H., Han, J. H., Kim, Y. H., Choi, S. H., & Kim, E. S. (2012). Preference-based solution selection algorithm for evolutionary multiobjective optimization. *Ieee Transactions on Evolutionary Computation*, *16*(1), 20–34.

Ozcan, U., & Toklu, B. (2009). Multiple-criteria decision-making in two-sided assembly line balancing: A goal programming and a fuzzy goal programming models. *Computers & Operations Research*, *36*(6), 1955–1965. doi:10.1016/j.cor.2008.06.009.

Park, S., Lee, T. E., & Sung, C. S. (2010). A three-level supply chain network design model with risk-pooling and lead times. *Transportation Research Part E-Logistics and Transportation Review*, *46*(5), 563–581. doi:10.1016/j.tre.2009.12.004.

Parmee, I., Cvetkovic, D., Bonham, C., & Packham, I. (2001). Introducing prototype interactive evolutionary systems for ill-defined, multi-objective design environments. *Advances in Engineering Software*, *32*(6), 429–441.

Parmee, I. C., Cvetković, D. C., Watson, A. H., & Bonham, C. R. (2000). Multiobjective satisfaction within an interactive evolutionary design environment. *Evolutionary Computatio*, *8*(2), doi:10.1162/106365600568176.

Rachmawati, L., & Srinivasan, D. (2010). Incorporation of Imprecise Goal Vectors into Evolutionary Multi-Objective Optimization. In *Paper presented at the 2010 IEEE World Congress on Computational Intelligence*. Spain: Barcelona.

Rajabalipour Cheshmehgaz, H., Desa, M., & Wibowo, A. (2011). A flexible three-level logistic network design considering cost and time criteria with a multi-objective evolutionary algorithm. *Journal of Intelligent Manufacturing (first online published)*, 1–17, doi:10.1007/s10845-011-0584-7.

Rajabalipour Cheshmehgaz, H., Desa, M. I., & Wibowo, A. (2012a). Effective local evolutionary searches distributed on an island model solving bi-objective optimization problems. *Applied Intelligence (first online published)*. doi:10.1007/s10489-012-0375-7.

Rajabalipour Cheshmehgaz, H., Desa, M. I., & Wibowo, A. (2012b). An effective model of multiple multi-objective evolutionary algorithms with the assistance of regional multi-objective evolutionary algorithms: VIPMOEAs. *Applied Soft Computing (first online published)*. doi:10.1016/j.asoc.2012.04.027.

Sato, H., Aguirre, H. E., & Tanaka, K. (2007). Local dominance including control of dominance area of solutions in MOEAs. In 2007 Ieee symposium on computational intelligence in multi-criteria decision making, 310–317, 402.

Shen, X. N., Guo, Y., Chen, Q. W., & Hu, W. L. (2010). A multi-objective optimization evolutionary algorithm incorporating preference information based on fuzzy logic. *Computational Optimization and Applications*, *46*(1), 159–188.

Stalk, G., & Hout, T. M. (1990). Competing against Time. *Research-Technology Management*, *33*(2), 19–24.

Suo, M. Q., Li, Y. P., & Huang, G. H. (2011). An inventory-theory-based interval-parameter two-stage stochastic programming model for water resources management. *Engineering Optimization*, *43*(9), 999–1018.

Tan, K. C., Khor, E. F., Lee, T. H., & Sathikannan, R. (2003). An evolutionary algorithm with advanced goal and priority specification for multi-objective optimization. *Journal of Artificial Intelligence Research*, *18*, 183–215.

Thiele, L., Miettinen, K., Korhonen, P. J., & Molina, J. (2009). A preference-based evolutionary algorithm for multi-objective optimization. *Evolutionary Computation*, *17*(3), 411–436.

Timm, G., & Herbert, K. (2009). *Comprehensive logistics*. London: Springer.

Vidal, C. J., & Goetschalckx, M. (1997). Strategic production-distribution models: A critical review with emphasis on global supply chain models. *European Journal of Operational Research*, *98*(1), 1–18.

Yang, W., Li, L., & Ma, S. (2007). Coordinating supply chain response-time: A bi-level programming approach. *International Journal of Advanced Manufacturing Technology*, *31*(9–10), 1034–1043. doi:10.1007/s00170-005-0260-1.

Zhang, Q., & Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *Ieee Transactions on Evolutionary Computation*, *11*(6), 712–731.

Zitzler, E., & Kunzli, S. (2004). Indicator-based selection in multiobjective search. *Parallel Problem Solving from Nature—Ppsn Viii*, *3242*, 832–842.

Zitzler, E., Laumanns, M., & Thiele, L. (2001). *SPEA2: Improving the strength pareto evolutionary algorithm*. Zurich, Switzerland: Swiss Federal Institute of Technology (ETH) Zurich.

Zitzler, E., Thiele, L., & Bader, J. (2008). SPAM: Set preference algorithm for multiobjective optimization. *Parallel Problem Solving from Nature—Ppsn X, Proceedings*, *5199*(847–858), 1164.