Simple heuristics-based selection of guides for multi-objective PSO with an application to electrical distribution system planning

N.C. Sahoo*, S. Ganguly, D. Das

Department of Electrical Engineering, Indian Institute of Technology, Kharagpur 721302, India

A R T I C L E  I N F O

Article history:
Received 26 April 2010
Received in revised form 3 January 2011
Accepted 3 February 2011
Available online 26 February 2011

Keywords:
Multi-objective particle swarm optimization
Selection of local guides
Pareto-optimality
Power distribution system planning
Cost-biased encoding

A B S T R A C T

In multi-objective particle swarm optimization (MOPSO), a proper selection of local guides significantly influences detection of non-dominated solutions in the objective/solution space and, hence, the convergence characteristics towards the Pareto-optimal set. This paper presents an algorithm based on simple heuristics for selection of local guides in MOPSO, named as HSG-MOPSO (Heuristics-based-Selection-of-Guides in MOPSO). In the HSG-MOPSO, the set of potential guides (in a PSO iteration) consists of the non-dominated solutions (which are normally stored in an elite archive) and some specifically chosen dominated solutions. Thus, there are two types of local guides in the HSG-MOPSO, i.e., non-dominated and dominated guides; they are named so as to signify whether the chosen guide is a non-dominated or a dominated solution. In any iteration, a guide, from the set of available guides, is suitably selected for each population member. Some specified proportion of the current population members follow their respective nearest non-dominated guides and the rest follow their respective nearest dominated guides. The proposed HSG-MOPSO is firstly evaluated on a number of multi-objective benchmark problems along with investigations on the controlling parameters of the guide selection algorithm. The performance of the proposed method is compared with those of two well-known guide selection methods for evolutionary multi-objective optimization, namely the Sigma method and the Strength Pareto Evolutionary Algorithm-2 (SPEA2) implemented in PSO framework. Finally, the HSG-MOPSO is evaluated on a more involved real world problem, i.e., multi-objective planning of electrical distribution system. Simulation results are reported and analyzed to illustrate the viability of the proposed guide selection method for MOPSO.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

The multi-objective optimization (MOO) has become a promising research area during the last decade as many real world optimization problems fall into this category. The major difficulty very often lies with the simultaneous optimization of the multiple conflicting objectives. No single solution can be an optimal solution; however, a set of solutions with a trade-off of the objectives represents the optimal solutions (Deb, 2001). The trade-off solutions are called the non-dominated solutions. The optimal set of the non-dominated solutions is known as the Pareto-optimal set from which a specific solution for an application can be chosen using a decision maker (DM). The MOO is traditionally based on three approaches, i.e., (i) a priori, (ii) interactive, and (iii) a posteriori. In the a priori approach, which is a preference-based strategy, comprising of relative preferences of different objectives is used. However, it is a difficult task to devise such a preference-based strategy without any knowledge of the likely trade-off solutions. Examples of the a priori approach are: weighting method, lexicographic ordering, goal-programming, etc. In the interactive approach, the DM articulates its preference during the optimization process and diverts the search direction towards its preferred zone where it intends to get the solution. However, it is extremely difficult for the DM to prefer a solution without knowing all the Pareto-optimal solutions. In the posteriori approach, the optimization aims at obtaining all the Pareto-optimal solutions from which the DM selects one as the final solution. Many MOO algorithms belong to this approach as there is no need of any prior knowledge about the relative importance of the objectives.

There are two approaches used for solving the MOO problems, i.e., classical search algorithm and evolutionary algorithm (EA). A classical search algorithm uses a point-by-point search method, where one solution in every iteration is evaluated and updated to obtain a better solution. The ultimate outcome of these methods is a single solution. On the other hand, in EA, the improvement of candidate solutions in successive iterations is essentially done by using point-to-point search approach utilizing collective
experiences from a group or neighborhood, since EA uses a population of potential candidate solutions. Thus, it yields multiple near-optimal solutions in its final population. The multi-point search ability to obtain multiple solutions in a simulation run makes EAs unique in solving MOO problems. There are different types of EAs, for example, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Tabu Search (TS), and Artificial Immune System (AIS). Among these, the GA has been mostly used for the MOO. The recent literature survey indicates the PSO to be a potential competitor of the GA (Kennedy and Eberhart, 1995; Shi and Eberhart, 1998; Bratton and Kennedy, 2007; Valle et al., 2008). The most common population based EAs are motivated from the evolution mechanism of nature. PSO is motivated from the simulation of social behavior of a flock birds, a fish school, etc. Although it cannot be concluded that PSO performs better than GA in general; PSO is reported to have many advantages, for example, easy implementation, effective memory use, and an efficient maintenance of the solution diversity (Bratton and Kennedy, 2007; Valle et al., 2008).

The main goals of multi-objective PSO (MOPSO) are to reach closer to the set of Pareto-optimal solutions (i.e., better convergence) and to get a set of diversified solutions (i.e., better diversity among the solutions) (Sierra and Coello Coello, 2006). Thus, the research on the MOPSO is mostly aimed at simultaneous attainment of these two goals. A better convergence can be achieved by selecting a suitable guide for each particle in the particle swarm. Unlike a very preliminary work (Parsopoulos and Vrahatis, 2002), all other works (Hu and Elbert, 2002; Hu et al., 2003; Mostaghim and Teich, 2003a; 2003b; Li, 2003; Coello Coello et al., 2004; Alvarez-Benitez et al., 2005; Zhang and Xue, 2007; Mostaghim et al., 2007; Agrawal et al., 2008a; 2008b; Durillo et al., 2009) follow the local/neighborhood best guide selection mechanism. The various types of local guide selection mechanisms are: dynamic neighborhood method (Hu and Elbert, 2002; Hu et al., 2003), Sigma method (Mostaghim and Teich, 2003a; 2003b), non-dominated sorting PSO (Li, 2003), roulette wheel selection and random selection (Coello Coello et al., 2004; Alvarez-Benitez et al., 2005), and sub-swarm based method (Zhang and Xue, 2007; Mostaghim et al., 2007; Agrawal et al., 2008a; 2008b). A better diversity among the solutions can be achieved either by using additional EA operators, i.e., mutation/turbulence operator (Mostaghim and Teich, 2003a; Coello Coello et al., 2004; Agrawal et al., 2008a) or by proper archiving of the set of elite/non-dominated solutions in successive iterations. There are various archiving techniques, for example, niche count (Li, 2003), crowding distance assignment (Li, 2003), e-dominance (Mostaghim and Teich, 2003b), and clustering technique (Zhang and Xue, 2007; Mostaghim et al., 2007; Agrawal et al., 2008b). However, the use of additional EA operators or special archive techniques imposes additional computational burden. There is also another powerful elitist MOO algorithm, i.e., Strength Pareto Evolutionary Algorithm-2 (SPEA2) used with GA (Zitzler et al., 2001), which can be used as a potential variant of MOPSO (Ganguly et al., 2009).

In this work, an attempt is made to improve the performance of MOPSO by a simple and efficient selection of guides using heuristics. The main idea behind the proposed heuristics-based guide selection mechanism for the MOPSO (HSG-MOPSO) is to utilize the influence of some specific dominated solutions obtained during the search. The HSG-MOPSO uses a novel approach for creation of a set of potential guides from the available non-dominated and dominated solutions in every iteration. The non-dominated guides are the best solutions (with respect to the non-dominance property) among the members in current iteration and are stored in an elite archive. The dominated local guides are heuristically chosen from the dominated solutions near the close proximity (region-wise in the objective space) of the Pareto-approximation front where no non-dominated solutions are available. In any iteration, certain specified percentage of the current population members follow their respective nearest non-dominated guides and the rest follow their respective nearest dominated guides. The idea is to maintain the search in all regions near the close proximity of Pareto-approximation front and also to enforce the search in those areas where no non-dominated solutions are available. Thus, it helps in keeping the diversity among the solutions without any additional EA operator or specific archiving technique.

In this paper, firstly, the performance evaluation of the proposed HSG-MOPSO is carried out on some multi-objective benchmark problems. The performances are also compared with those of the Sigma-MOPSO and the SPEA2-MOPSO and assessed by several statistical tests and time-complexity analysis. The effects of the controlling parameters of the HSG-MOPSO have been investigated and reported. Subsequently, the HSG-MOPSO is applied on a real world problem, i.e., multi-objective planning of electrical distribution systems. This is a multi-objective optimization problem involving a number of planning/design variables (Khator and Leung, 1997) such as: (i) number of feeders and their routes, (ii) number and location of sectionalizing switches, and (iii) type of network structure, i.e., radial or meshed. These planning decisions are obtained by optimization of multiple objectives such as: (i) minimization of the installation cost of the new facilities (i.e., substations and feeders), (ii) minimization of the operational (maintenance and lost energy) cost, and (iii) maximization of the system reliability. The network planning must also meet several constraints such as substation and feeder capacity limits, maximum node voltage deviation, etc. Several solution strategies for this problem are already reported in the literature, for example, non-linear programming (Fletcher and Strunz, 2007a; 2007b), dynamic programming (Nahman and Spiric, 1997; Boulaxis and Papadopoulos, 2002), network flow programming (Tang, 1995), Benders' decomposition (Kagan and Adams, 1993), GA (Miranda et al., 1994; Ramirez-Rosado and Bernal-Agustin, 2001; Carrano et al., 2006; Mendoza et al., 2006; Rivas-Davals and Irving, 2005), tabu search (Ramirez-Rosado and Dominguez-Navarro, 2006), artificial immune system (Carrano et al., 2007), PSO (Ganguly et al., 2010), etc. In this study, the efficacy of the HSG-MOPSO on this planning optimization problem is investigated.

The organization of this paper is as follows. A brief overview of MOPSO is provided in Section 2. The development of the HSG-MOPSO is presented in detail in Section 3 along with its performance evaluation on different benchmark MOO problems. In Section 4, a real world application of the HSG-MOPSO, i.e., multi-objective planning of electrical distribution systems is reported. Section 5 concludes the paper. The appendix includes the list of principal symbols used in this paper and the main computational steps of the SPEA2-MOPSO and Sigma-MOPSO.

2. Multi-objective PSO (MOPSO): a brief overview

Particle Swarm Optimization (PSO) is a population based multi-point search technique (Kennedy and Eberhart, 1995) that mimics the social behavior of a flock of birds, a fish school, etc. Like any other EA, the search starts with a population of search points called particles. Each particle is encoded by a (continuous) position vector (X), containing multi-dimensional information (initially chosen at random), which is updated by using particle's velocity (initially chosen at random) in successive iterations. The velocity vector (PV) of a particle is updated using its own previous best position (pbest) and the best neighborhood particle's position.
Higher inertia constitutes a global search by particle velocity is updated by using Eq. (3) where the inertia facilitates local search. In this varying inertia model, the weight is multiplied with the velocity of previous iteration of a particle

\[
P_{\text{iter}}^{\text{iter}+1} = wP_{\text{iter}}^{\text{iter}} + \phi_1r_1(p_{\text{best iter}}^{\text{iter}} - X_{\text{iter}}^{\text{iter}}) + \phi_2r_2(n_{\text{best iter}}^{\text{iter}} - X_{\text{iter}}^{\text{iter}})
\]

(1)

Among the three component terms in Eq. (1), the first term is referred to as momentum or habit, i.e., the tendency of a particle to continue in the same direction it has been traveling. The second and third terms, which are scaled by two learning constants \((\phi_1, \phi_2)\) and two random numbers \((r_1, r_2)\), are referred to as guidance by self-knowledge and group knowledge, respectively (Valle et al., 2008). The position of a particle is updated with its current velocity using Eq. (2). The fitness of a particle is determined by a pre-defined objective function. A particle's \(n_{\text{best}}\) acts as a guide and the best neighbor is judged by the fitness evaluation. PSO is carried out iteratively till some specified termination criterion is attained, i.e., specified maximum number of iterations or a certain desired particle fitness. Several modifications to this basic PSO have also been reported (Shi and Eberhart, 1998; Bratton and Kennedy, 2007; Valle et al., 2008). Among them, an efficient modification is that of a linearly decreasing inertia weight \((w)\) for balanced local and global search (Shi and Eberhart, 1998). Higher inertia constitutes a global search by putting more weights on the previous experience and a lower inertia facilitates local search. In this varying inertia model, the particle velocity is updated by using Eq. (3) where the inertia weight is multiplied with the velocity of previous iteration of a particle

\[
P_{\text{iter}}^{\text{iter}+1} = wP_{\text{iter}}^{\text{iter}} + \phi_1r_1(p_{\text{best iter}}^{\text{iter}} - X_{\text{iter}}^{\text{iter}}) + \phi_2r_2(n_{\text{best iter}}^{\text{iter}} - X_{\text{iter}}^{\text{iter}})
\]

(2)

The performance of the PSO depends a lot on the information exchange among the particles, which is achieved by using various neighborhood topologies. The most common neighborhood topologies are the global best \((gbest)\) and local best \((ibest)\) topologies (Bratton and Kennedy, 2007). In the global topology, each particle shares information with all other particles instantaneously resulting in initial faster convergence, but with a great possibility of being trapped in local minima. In the ibest topology, a particle only shares its experience with its immediate neighbors. In a ring neighborhood, each particle has two neighbors.

PSO has been extended for solving the MOO problems, which is appropriately called the MOPSO. There are various approaches for the MOPSO (Sierra and Coello Coello, 2006), for example, the aggregating approach, lexicographic ordering approach, sub-population approach, and Pareto-based approach. Among these the Pareto-based MOPSO, where the Pareto-dominance principle is directly used to optimize all the objectives simultaneously, is very much popular (Sierra and Coello Coello, 2006). The obvious advantage of the Pareto-based approaches over other non-Pareto-based approaches is that the Pareto-dominance principle is used to guide the search. Specifically, the selection of guides is based on the Pareto-dominance concept, which helps to obtain efficient solutions. However, the non-Pareto-based approaches have much simpler and straightforward implementations. This work is focused on the Pareto-based MOPSO.

2.2. Guide selection mechanisms for MOPSO

As noted earlier, the two main goals of the Pareto-based MOPSO are to obtain better convergence and better diversity among the solutions. There are various approaches to achieve these goals (Sierra and Coello Coello, 2006). The guide selection, i.e., determination of appropriate \(n_{\text{best}}\) particle so as to modify the third component of Eq. (1), is one of such approaches. The objective of this work is to develop a new guide selection mechanism that can alleviate some of the problems associated with the existing guide selection methods. For this purpose, two well-established guide selection methods, i.e., SPEA2 and Sigma methods are chosen. The philosophy behind these two methods is given below and their computational steps are given in the Appendix.

2.2.1. Guide selection using SPEA2

Since there are multiple objectives in a MOO problem, none of the objectives can be exclusively used as the fitness of a particle. In SPEA2, a fitness value is assigned to each particle based on its non-domination strength and the density around it (Zitzler et al., 2001) as given in the Appendix. This type of fitness assignment scheme is required so that fitness of a particle becomes a reflection of the multi-objectivity of a MOO problem. In the SPEA2-MOPSO, an elite archive is created to preserve the non-dominated solutions found by the optimization algorithm and then it is used to assign fitness to each member of the archive itself and the current population undergoing evolution. The guide selection in the SPEA2-MOPSO is done with the fitness value of each particle by using any PSO neighborhood topology. For example, in global best topology, the best particle chosen among all the particles according to fitness is considered to be the guide for all the particles. In local best topology, the guide for a particle is chosen from its neighboring particles according to the fitness values.

2.2.2. Guide selection using Sigma method

The Sigma method is another guide selection method requiring less computation. A Sigma value is assigned to each particle based on the objective functions (Mostaghim and Teich, 2003a) as given in the Appendix. Two particles with closer Sigma values are basically located on two lines (with respect to the origin) which are closer to each other in the objective space. All non-dominated solutions are considered as candidate guides and they are stored in an elite archive. The guide for a particle is the member of the elite archive with the closest Sigma value. The idea behind this method is similar to compromise programming (CP), i.e., to direct all the particles towards the ideal/utopia (Pareto-optimal) points as in CP.

3. Heuristics-based guide selection for MOPSO (HSG-MOPSO)

There are some inherent difficulties generally associated with MOPSO, i.e., loss of diversity in search and convergence in the multi-frontal problems (Durillo et al., 2009). To emphasize this, the difficulties that may come up in some typical situations while using the Sigma- and the SPEA2-based selection of guides in MOPSO are examined. Then the HSG-MOPSO is developed to address these situations. Without loss of generality, a general bi-objective minimization problem is considered to illustrate the basic ideas. The performance evaluation on few MOO benchmark problems is also presented and analyzed.
3.1. Some difficulties with the SPEA2 and Sigma method

In any form of PSO, the motion of a particle is influenced by a guide chosen from the population. However, unlike the search for a single target in the mono-objective optimization (Fig. 1(a)), a multi-objective optimization needs a search for multiple targets to reach closer to the Pareto-optimal front (Fig. 1(b)). In MOPSO using the SPEA2- and Sigma method based selection of guides, there might be problems in some specific scenarios as described below.

In the SPEA2-MOPSO, certain fitness is assigned to the particles based on the Pareto-dominance and solution density. Then, the guide selection (for a particle) is done by following the best/ring topology, in which a particle shares information with its immediate neighbors as shown in Fig. 2(a) and it follows the fitter particle among its two immediate neighbors. The global best topology is not used in this work as the search in the global best topology is aimed towards single target for all particles in an iteration, which may cause loss of diversity among solutions. In best topology, a search for multiple targets can be obtained; but the search directions of all the particles may not always be towards the Pareto-optimal front. A typical such case is shown in Fig. 2(b) where a particle follows a dominated solution; for instance, particle 7 follows particle 6. Moreover, the search directions of all the particles may frequently change in different iterations which may prevent them from reaching closer to the Pareto-optimal front. For a problem where the two end regions of the Pareto-optimal (approximation) front are very much flat (Fig. 2(c)), a typical situation may arise when the difference between one objective function for two neighboring solutions (say, solutions 3 and 4) is very high compared to the difference between the other objective function in either end regions of the Pareto-front. Thus, a little improvement of one objective of a solution (from solution 3 to solution 3’) may cause domination of the other solution (solution 4). As soon as any solution gets dominated, the search direction is most likely diverted away from it. This prevents the chances of improvement of the solutions in those areas and those specific regions of the Pareto-approximation front may remain unexplored. This situation may also appear in an m-objective optimization if the difference between two adjacent non-dominated solutions in terms of \((m-1)\) objectives is less and the difference in terms of one objective is high.

In the Sigma-MOPSO, as mentioned earlier, a particle follows the nearest elite/non-dominated solution (Fig. 3(a)) with the closest Sigma value. Hence, there is no chance of the selection of dominated guides in the Sigma-MOPSO. This is a distinct difference between the Sigma-MOPSO and SPEA2-MOPSO. A drawback of this method surfaces if, in any iteration, one solution dominates all other solutions as shown in Fig. 3(b). In this condition, the diversity of the search is lost as the search is aimed towards single target.

In view of the deficiencies with the SPEA2- and Sigma method, the HSG-MOPSO is proposed.

3.2. HSG-MOPSO

The HSG-MOPSO is motivated by the mechanism of the SPEA2-MOPSO with best topology in which some dominated solutions can be potential guides. However, the dominated guides are purposefully selected and utilized in the HSG-MOPSO so as to obtain better diversity among the solutions. On the contrary, the dominated solutions may be selected as guides only in case of some situations, by chance, in SPEA2-MOPSO as illustrated in Fig. 2(b). The Pareto-optimal front which may be convex, non-convex or disconnected, is in general not known a priori. The HSG-MOPSO has good ability of keeping diversity as the guides are selected from both non-dominated and some better dominated solutions (chosen from the close proximity (strip-wise in the objective space) of the Pareto-approximation front where no non-dominated solutions are available). The guides are selected for the particles so as to put more efforts to discover the non-dominated solutions in those areas. A typical two-objective Pareto-approximation
3.2.1. Creation of the set of potential guides

The non-dominated solutions are represented by squares. The non-dominated solutions are missing in some areas; some front is shown in Fig. 4 to illustrate this process; the blank areas (where all solutions are dominated).

Selection of guide (from the set of potential guides) for each particle from the strip.

(i) The objective space is divided into a number of vertical and horizontal strips along the X-axis (say, X-1 to X-10) and Y-axis (say, Y-1 to Y-10), respectively, as shown in Fig. 4.

(ii) The candidate guides are chosen from these vertical and horizontal strips separately. To do this, firstly, a search/scanning for the non-dominated solutions is performed in all the strips along the X-axis. A non-dominated solution appearing in any strip is chosen as a potential guide; it is accordingly named as a non-dominated guide. If no non-dominated solution exists in any strip, then the dominated solution closest to the Pareto-approximation front is considered as a potential guide and it is accordingly named as a dominated guide. Unless any strip is completely empty, at least one solution, either a non-dominated or dominated one, is chosen as a guide from the strip. The same procedure is repeated along the Y-axis. All the candidate guides are stored in a set for further use.

For example, in Fig. 4, the non-dominated solutions are missing in the strips X-6, X-7, and X-8. Thus, the solutions 1, 3, and 5 are chosen as guides for these strips, respectively. Similarly, solutions 6 and 8 are chosen as guides for strips Y-8 and Y-7, respectively. These are called the dominated guides.

3.2.2. Guide selection for each particle

The main objective behind choosing a non-dominated guide for a particle is to search the neighborhood of the guide-set, all the particles are assigned with the non-dominated guides.

3.3. Performance evaluation of the HSG-MOPSO on multi-objective benchmark problems

The performance of the HSG-MOPSO is validated on some typical benchmark optimization problems with different complexity levels, for example, convex, non-convex, continuous, disconnected, multi-frontal, Boolean Pareto-fronts, etc. The chosen test problems are: six problems from ZDT series (Deb, 2001), Kursawe test problem (KUR) (Deb, 2001), OKA2 (Okabe et al., 2004). The performance comparisons with the SPEA2-MOPSO and the Sigma-MOPSO are presented. In this investigation, no mutation/turbulence operator is used for proper assessment of the influence of different guide selection methods on the convergence and diversity directly. For the problem consisting of binary variables only, i.e., ZDT5, the position updating is done by binary-PSO (BPSO) (Kennedy and Eberhart, 1997), which uses...
logical transformation of the velocity information using the sigmoid function.

3.3.1. Optimization of PSO parameters

The PSO parameters are empirically/experimentally optimized by exhaustive simulation studies for all the benchmark problems1. The optimized values are given in Table 1 ($d_{\text{max}}, d_{\text{min}}$ are the maximum and minimum values of the problem variables).

3.3.2. Performance assessment

The Pareto-approximation fronts for two benchmark problems, for which, there exists some noticeable differences among the performances of the three MOPSO algorithms are shown in Figs. 6 and 7. It is observed that Sigma-MOPSO converges to a single point for ZDT2 (Fig. 6) and it also fails to discover some portions of the disconnected Pareto-approximation front for ZDT3 as shown in Fig. 7. This happens due to lack of diversity in the search process of the Sigma-MOPSO. However, the performances of the HSG-MOPSO and SPEA2-MOPSO appear to be more or less same.

This is further investigated with the results of 25 independent runs using three well-known performance measurement indicators (Deb, 2001; Knowles et al., 2006), i.e., hypervolume indicator, diversity indicator, and summary attainment surface plots.

(i) Hypervolume Indicator: It is a unary indicator (Deb, 2001) used to measure the portion of objective space dominated by the Pareto-approximation set. A higher hypervolume indicator signifies larger area dominated by the approximation set indicating better convergence. The means and variances of the hypervolume indicator for different test problems with the HSG-MOPSO, SPEA2-MOPSO, and Sigma-MOPSO are shown in Table 2. The corresponding reference points for measuring the hypervolume indicator for different benchmark problems are given in Table 2.

---

1 Same empirical optimization process is followed for optimization of the PSO parameters for all the test problems as well as for the distribution system planning problem. In order to put more emphasis on the real-world engineering application problem, the sequential steps of the empirical optimization process is provided for the distribution system planning problem in Section 4.3.1.
Remark 1. The results show that the performances of the HSG-MOPSO and SPEA2-MOPSO are very much comparable in most of the test problems and they are significantly better than those of the Sigma-MOPSO in many test problems, i.e., ZDT1, ZDT2, ZDT4, and OKA2. In ZDT4 and KUR, the HSG-MOPSO achieves comparatively better convergence than the SPEA2-MOPSO.

(ii) Diversity indicator: The diversity indicator (Deb, 2001) \( \mathcal{D} \), defined in Eq. (5), measures both diversity and spread among the non-dominated solutions.

\[
\mathcal{D} = \frac{\sum_{i=1}^{M} d_i^2 + \sum_{j=1}^{N_{ndf}} |d_j - \bar{d}|}{\left( \sum_{i=1}^{M} d_i^2 + N_{ndf} \bar{d} \right)}
\]  

Ideally, the distance between the two nearest solutions should be equal to the mean distance between any solution pair and the distances between the extreme solutions of the Pareto-approximation sets and the Pareto-optimal sets should be zero. So, the diversity metric should ideally be zero. The means and variances of the diversity indicators for the Pareto-approximation sets are shown in Table 2.

Remark 2. The diversity indicators show that the performance of the HSG-MOPSO and the SPEA2-MOPSO are mostly similar except for the test problems with disconnected Pareto-fronts. The HSG-MOPSO exhibits much better diversity in these test problems. The performance of the HSG-MOPSO is mostly better than that of the Sigma-MOPSO.

(iii) Summary attainment surface plot: As the solutions in an approximation set cannot be interpolated, it is difficult to visualize/assess the quality of approximation sets. One of the performance assessment techniques is to plot the summary attainment surface from multiple trial runs. A summary attainment surface is a boundary comprising of all the tightest goals attained by a particular optimizer after a specified number of trial runs (Knowles et al., 2006). The tightest goals after certain number of trial runs are the resultant non-dominated solutions amongst all the Pareto-approximation sets. The summary attainment surface plots after 25 runs obtained with the three MOPSO algorithms are shown in Fig. 8 for each test problem.

Remark 3. The attainment surface plots for the HSG-MOPSO and SPEA2-MOPSO are mostly indistinguishable except for ZDT4 and ZDT5 where better attainment is obtained with the HSG-MOPSO. The attainment surfaces for the Sigma-MOPSO in ZDT2 and ZDT4 are single points because the Sigma-MOPSO is unable to hold the diversity. On the contrary, the HSG-MOPSO and the SPEA2-MOPSO perform consistently without losing diversity.

In Fig. 9, the percentage attainment surfaces obtained with the HSG-MOPSO for two test problems are shown. The \( \mu - \% \) attainment surface consists of the goal vectors likely to be attained with \( \mu - \% \) of trial runs. The goal vectors are the resultant non-dominated solutions among all the Pareto-approximation sets obtained with \( \mu - \% \) of trial runs. The performance of any optimizer
is said to be good if all the tightest goals (i.e., resultant non-dominated solutions with 100% trial runs) can be attained with less trial runs (i.e., lower $\mu$). The results show little difference between percentage attainment surfaces obtained with 50% and 100% trial runs. This illustrates the goals obtained with 50% trial runs are very close to the tightest goals. Thus, the performance of the HSG-MOPSO is good.

### 3.3.3. Sensitivity of number of strips per objective

Since the number of strips per objective is a control parameter in the HSG-MOPSO, a sensitivity test is required to assess the influence of this parameter. A study is carried out by taking 25 independent runs for some selected test problems. The variations in the mean hypervolume and diversity indicators with the number of strips per objective for three test problems, i.e., ZDT1, ZDT5, and ZDT6 are shown in Fig. 10. The results illustrate that the convergence characteristics of the HSG-MOPSO, as measured by the hypervolume indicator, is less sensitive to the number of strips per objective. However, the diversity of the solutions measured by the diversity indicator is very much sensitive to the number of strips per objective. It is also observed that the diversity improves with increasing number of strips per objective up to a certain point (say 20–25) beyond which no significant improvement occurs. Obviously, it is a problem specific parameter.

### 3.3.4. Impact of the parameter $K$ in the HSG-MOPSO

An empirical study is also performed to observe the impact of the parameter $K$ on the convergence characteristics and the diversity among solutions. As noted earlier, a higher value of $K$ helps in convergence towards Pareto-optimal front and a lower value of $K$ promotes diversity among the solutions. These facts are corroborated from simulation experiments. The convergence and diversity achieved by the HSG-MOPSO for different values of $K$ are measured by mean hypervolume and mean diversity indicators, respectively, over 25 independent runs. The results obtained with three relatively difficult test problems are shown in Fig. 11. There is a clear trend for better convergence with increasing values of $K$. This implies that better convergence can be obtained if more particles follow the non-dominated guides. On the other hand, the diversity among the solutions becomes worse with increasing values of $K$. This illustrates that better diversity can be obtained with lower value of $K$, i.e., when more particles follow their dominated guides.

### 3.3.5. Variations of number of dominated guides with HSG-MOPSO iterations

In principle, the number of dominated guides should decrease as the algorithm approaches convergence towards the Pareto-optimal front. This effect can be clearly seen in Fig. 12 which displays the number of dominated guides vs. HSG-MOPSO iterations for three chosen test problems. For ZDT1 (continuous Pareto-optimal front), the number of dominated guides becomes zero within 70 iterations. An initial increasing trend of the number of dominated guides is observed for ZDT5 (Boolean function problem). Then, it sharply decreases and becomes zero around 130 iterations. For OKA2, the number of dominated guides slowly decreases compared to the other two test problems as the Pareto-optimal front is disconnected. These observations point to the fact that missing areas (i.e., no non-dominated solutions) are really created in the Pareto-approximation front and the non-dominated solutions are gradually obtained in those areas as the iteration number increases justifying the usefulness of the dominated guides in the search.
3.4. Run-time complexity

The execution speed of an algorithm is measured by its run-time complexity. It is the worst-case running time required to execute a task and is represented by the $O$-notation. In MOPSO, the run-time complexity is governed by two factors, i.e., (i) velocity and position updating scheme in an iteration and (ii) archive truncation. In Table 3, a comparison of the run-time complexities of the HSG-MOPSO, SPEA2-MOPSO, and Sigma-MOPSO is provided.

- In the HSG-MOPSO, the run-time for velocity and position updating of a particle largely depends on the creation of the set of potential guides and the selection of guide for each particle, whose run-time complexities are $O(N_S N_T)$ and $O(N_{S} N_{G})$, respectively. Thus, the total run-time complexity is a quadratic function. The run-time complexity of archive truncation is $O(N_{S}^{2} \log N_{A})$.
- In the SPEA2-MOPSO, fitness is assigned to each particle and the velocity and position of a particle are updated by comparing its fitness with the two immediate neighboring particles. Thus, the run-time complexity of the SPEA2-MOPSO depends on the fitness assignment which has two parts, i.e., raw fitness assignment based on the Pareto-dominance and the nearest neighborhood density estimation (Zitzler et al., 2001). The run-time complexities of these two tasks are $O(N_{S}^{2} T)$ and $O(N_{S}^{2} \log N_{p})$, respectively. The run-time complexity for archive truncation is also $O(N_{S}^{2} \log N_{A})$.
- In the Sigma-MOPSO, the run-time complexity depends on the selection of local guides from the archive. This needs $O(N_{L} N_{A})$. 

![Fig. 8. Summary attainment surface plots for: (a) ZDT1 normalized with (1,1), (b) ZDT2 normalized with (1,1), (c) ZDT3 normalized with (1,1), (d) ZDT4 normalized with (1,10), (e) ZDT5 normalized with (30,30), (f) ZDT6 normalized with (3,1), (g) KUR normalized with (20,11.5), and (h) OKA2 normalized with (3,1).]
computations in any iteration. The run-time complexity for the archive truncation is same as that of the SPEA2-MOPSO.

Out of these three algorithms, the SPEA2-MOPSO has the highest run-time complexity for velocity and position updating processes, whereas the HSG-MOPSO and Sigma-MOPSO possess more or less similar run-time complexities. However, all the three algorithms have similar run-time for the archive truncation.

4. Multi-objective planning of electrical distribution systems using HSG-MOPSO

This section deals with an engineering application of the HSG-MOPSO, i.e., multi-objective electrical distribution network planning. There are two types of planning. They are (i) static planning, i.e., planning of a completely new network and (ii) expansion planning, i.e., addition of nodes in an existing network. The planning model used in this work follows the formulation given in [Ganguly]...
et al., 2010). In addition, a flexibility to generate both radial and meshed structures has been incorporated into the planning.

4.1. Multi-objective planning model for distribution systems

In this model, there are two optimization objectives, i.e., construction of economical and reliable network. The first objective deals with minimization of the total installation and operational cost, while the second objective is aimed at maximization of the reliability. The objective function-1 consists of the installation costs of the new facilities (substations, feeder, and sectionalizing switches) plus the maintenance costs of substations and feeders and the cost of energy losses. The objective function-2 measures network reliability with an index called the contingency-load-loss ratio (CLLI) which is defined as the ratio of the average non-network reliability with some (indirect) guiding information about the solution. A particle representing a distribution network carries following information:

(i) Node bias values ($\rho$): Each node of the distribution network is assigned one dimension in the position vector and the corresponding parameters representing the various nodes are called the node bias values, i.e., $\{\rho \in (-1,0,1,0)\}$, which when decoded appropriately generates a radial network structure. This is the indirect information.

(ii) Number of feeders ($\eta_f$): This occupies another dimension of the position vector providing direct information on the number of feeders in the network. The number of feeders is kept within a specified maximum value.

(iii) Number of sectionalizing switches ($\eta_{sw}$): This is another dimension providing direct information on the number of sectionalizing switches in the network. The number of sectionalizing switches is kept within a specified maximum value.

(iv) Number and location (zone) of the meshes/loops ($\eta_{loop}$): This segment contains the information on the number and locations of the loops. A novel procedure is followed to form the loops. First, the whole network area is divided into a number of zones as shown in Fig. 13(b). A node is kept in a zone depending on its geographical location. The total number of zones is problem specific. In case of a meshed network (i.e., $\eta_{loop} \neq 0$), first, a radial network is formed from the available utility. The conductor, substation sizes, and circuit breaker types are chosen from a set of pre-defined sizes. The breaker types are decided based on the short circuit capacity of the networks (Chen and Huang, 2009). The minimization of the CLLI improves system reliability. As the meshed networks have better reliability than the radial networks, they have lower CLLI. Also, the inclusion of more sectionalizing switches in a network reduces its CLLI. The factors that influence both the objectives simultaneously are:

- **Network structure**: A meshed network is more reliable as it provides alternative supply paths within a mesh at the expense of additional branch installation cost. Moreover, since the short circuit capacity level of a meshed network is higher than that of a radial network with similar feeder routing (Chen and Huang, 2009), the circuit breakers and auxiliary switch-gears are more costly.

- **Number of feeders**: Multiple feeders improve network reliability (i.e., reduces the CLLI), but it increases the installation cost due to the presence of additional circuit breakers.

- **Sectionalizing switch**: It improves the objective function-2; but its installation cost increases the objective function-1.

Thus, the simultaneous minimization of these two objective functions obviously needs a trade-off analysis. This is done by using the Pareto-dominance principle (Deb, 2001). The optimization constraints are: substation/feeder capacity limits and node voltage limits.

4.2. Multi-objective planning algorithm for distribution systems

This subsection provides the multi-objective planning algorithm for electrical distribution systems. This algorithm consists of several components, for example, particle encoding/decoding scheme for distribution network representation, branch conductor size selection, and constraint handling techniques.

4.2.1. Particle encoding/decoding scheme

In this work, a particle position vector is encoded using a combination of two types of particle (position vector) encoding schemes, i.e., direct and indirect encoding as shown in Fig. 13(a). The former directly represents the solution, whereas the latter carries some indirect guiding information about the solution. A particle representing a distribution network carries following information:

\begin{equation}
\eta_{loop} = \frac{N_{DL_{loop}}}{L_{total}}
\end{equation}

The minimization of the objective function-1 optimizes the capital investment cost and the annual operational cost of a network. As the meshed networks have better reliability than the radial networks, they have lower CLLI. Also, the inclusion of more sectionalizing switches in a network reduces its CLLI. The factors that influence both the objectives simultaneously are:

- **Network structure**: A meshed network is more reliable as it provides alternative supply paths within a mesh at the expense of additional branch installation cost. Moreover, since the short circuit capacity level of a meshed network is higher than that of a radial network with similar feeder routing (Chen and Huang, 2009), the circuit breakers and auxiliary switch-gears are more costly.

- **Number of feeders**: Multiple feeders improve network reliability (i.e., reduces the CLLI), but it increases the installation cost due to the presence of additional circuit breakers.

- **Sectionalizing switch**: It improves the objective function-2; but its installation cost increases the objective function-1.

Thus, the simultaneous minimization of these two objective functions obviously needs a trade-off analysis. This is done by using the Pareto-dominance principle (Deb, 2001). The optimization constraints are: substation/feeder capacity limits and node voltage limits.

**Table 3**

Comparison of run-time complexities of the HSG-MOPSO, SPEA2-MOPSO, and Sigma-MOPSO.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Run-time for velocity and position updating scheme per objective in an iteration</th>
<th>Run-time for archive truncation per objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSG-MOPSO</td>
<td>$O(N_rN_f + N_{sw})$</td>
<td>$O(N_r^2 \log N_L)$</td>
</tr>
<tr>
<td>SPEA2-MOPSO</td>
<td>$O(N_r^2 N_f + N_{sw})$</td>
<td>$O(N_r^2 \log N_L)$</td>
</tr>
<tr>
<td>Sigma-MOPSO</td>
<td>$O(N_rN_f)$</td>
<td>$O(N_r^2 \log N_L)$</td>
</tr>
</tbody>
</table>

**Fig. 12.** Variation of number of dominated guides with HSG-MOPSO iterations.
Then, it is converted into a meshed network by creating \( \eta_{\text{loop}} \) number of loops. A loop is created by connecting any two nodes of the two selected zones, such as zones \( Z^l_1 \) and \( Z^e_1 \) as shown in Fig. 13(b). This segment provides direct information on the number and locations/zones of loops. The number of loops is restricted between zero (i.e., radial structure) and a certain maximum value.

The generalized particle decoding scheme, as shown in Fig. 13(c), consists of the following four steps.

**Step-1: Formation of a radial network modified cost-biased decoding scheme**

In this cost-biased decoding scheme, the nodes are selected and appended to the terminal node of a growing path on the basis of minimum value of the product of the branch costs and node bias values (Mohemmed and Sahoo, 2007)

\[
j = \arg\min \{C_{\text{br}}(i,l) \rho_k\} \quad C_{\text{br}}(i,l) = \text{branch cost}(\text{nodes } i \text{ and } l) \tag{8}\]

As a radial network has both main and lateral branches, the proposed decoding scheme is a modification of the cost-biased decoding for this problem (Mohemmed and Sahoo, 2007). Moreover, in a distribution network, it is generally desired to connect a node with few neighboring nodes only. This feature is incorporated by constructing a binary connectivity matrix \( B \). For a \( a \times a \) system, \( B \) is a \( a \times a \) matrix. If a connection between nodes \( i \) and \( j \) is allowed, \( B(i,j) = 1 \), else \( B(i,j) = 0 \). The maximum number of allowable connections for a node is problem specific. In this way, the total number of branches \( (A_{\text{br}}) \) involved in the optimization is chosen. Fig. 14 shows the pseudocodes of the decoding scheme to generate the network structure. The length of a branch is taken as its cost. At the start, all the nodes to be connected are stored in an array \( [R] \) and the substation node is stored in another array \( [Q] \). Two more arrays are created to store the start and end nodes of the branches. Initially they are empty for a static planning and they consist of start and end nodes of the existing network for an expansion planning. The number of nodes that are directly connected with a substation is restricted to the number of feeders \( (\eta_{\text{F}}) \) obtained from the particle and they are selected based on minimum distances from the substations. Those nodes are deleted from \( [R] \) and added to \( [Q] \); \( \{x_i\}, \{z_e\} \) are updated accordingly.

**Step-2: Formation of loops (for meshed network)**

In this step, the radial network obtained from Step-1 is converted into a meshed network. The zones, in which the start and end nodes of a loop-forming branch are located, are directly obtained from the particle. For example, the start and end nodes of the loop-forming branch of the first loop belong to the zones \( Z^l_1 \) and \( Z^e_1 \), respectively. Any two nodes from these two zones are connected to form the first loop.

**Step-3: Placement of sectionalizing switches**

The sectionalizing switches are placed after obtaining the network structure. It is done heuristically based on the number of switches obtained from the particle. The switch locations are obtained as follows:

- The number of switches in an individual feeder (say feeder \( F_i \)) is determined as:

\[
\eta_{\text{Sw}}^i = \text{round}(a_F \eta_{\text{Sw}}/a) \tag{9}\]
The sectionalizing switches are placed for improving the reliability that depends on two factors, i.e., (i) load demand at the nodes and (ii) number of branches in different feeder sections, sectionalized by the sectionalizing switches. Hence, two switch placement strategies are proposed. The first strategy is based on the load demand at different nodes and the second one is based on the number of branches/nodes in the individual feeders. The first strategy attempts to equalize the total load on different feeder sections. In the second strategy, the switches in a feeder are conveniently (and uniformly) placed at equal distances based on a heuristic judgment for keeping same number of branches in all the feeder sections as far as possible. For example, consider the case of a feeder serving 15 load nodes with a total demand of 1500 kW and two sectionalizing switches are to be placed in that feeder. In the first strategy, the two switches are placed such that the load demands of the three feeder sections are nearly 500 kW each. In the second strategy, one switch can be placed in the branch immediately following the 5th node and the other in the branch immediately following the 10th node so as to obtain equal number of branches (nearly 5) in all the three feeder sections.

4.2.2. Conductor size selection

The branch conductor sizes are selected based on the tentative branch power flow so as to keep the branch flows within the respective rated capacities. All the branches of a network are first assigned with arbitrary conductor sizes and a load flow is performed to get the power flow in all the branches. Thereafter, the actual conductor sizes, with ratings (immediately) higher than the respective branch flows, are assigned. Again, the load flow is performed and it is checked whether any branch flow exceeds its rating. If it exceeds in any branch, the same procedure is repeated and the process is continued till the ratings of all the branches are above their respective branch flows.

4.2.3. Constraint handling techniques

The constraints in this planning problem are handled as follows:

- The feeder branch capacity is satisfied by the proposed conductor size selection algorithm.
- If the node voltage limit constraint is violated, the solution is penalized by a suitable penalty factor to minimize its influence on other particles. The penalty factor is computed as the product of the absolute value of the maximum node voltage deviation from a specified nominal value and a high integer number. This penalty is added to the objective function-1.

The pseudocodes of the distribution system planning using the HSG-MOPSO are provided in Fig. 15.

4.3. Simulation results

The distribution system planning algorithm is evaluated via computer simulation studies on a 21-node system (Ganguly et al., 2009; Carrano et al., 2006) and a 54-node system (Ranjan et al., 2002). Some typical features of these systems are:

- The 21-node system has one existing substation and four existing branches to serve five load nodes. It is an expansion planning problem. The optimization plan is to connect rest of the 16 nodes with the existing network. The conductor sizes are taken from Carrano et al. (2006).
- The 54-node system has no existing network. It is a static planning problem. The optimization planning is to set up one substation in a given location to serve 53 load nodes. The substation capacity is 5 MVA which serves a total load demand of 2.18 MVA. The installation cost of the substation is $3,000,000. Two conductor sizes are used for this planning problem (Table 4).
- The circuit breaker specifications are given in Table 5.
- The minimum and maximum node voltage limits are taken as 0.92 and 1.08 p.u., respectively. The substation voltage is considered as 1.0 p.u.
- The installation cost of a sectionalizing switch is taken as $20,000.
- The maximum number of feeders and loops (both) are specified to be 3. The maximum number of allowable connections for a node is chosen as 8. The Pareto-approximation sets obtained for this planning are presented and some typical features are highlighted. The performance assessment for the HSG-MOPSO,

Table 4

<table>
<thead>
<tr>
<th>Conductor type</th>
<th>Current rating [A]</th>
<th>Branch installation cost ($/km)</th>
<th>Preventive maintenance cost ($/km/year)</th>
<th>Corrective maintenance cost ($/km/year)</th>
<th>Resistance (ohm/km)</th>
<th>Reactance (ohm/km)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>150</td>
<td>10,000</td>
<td>533.54</td>
<td>6.51</td>
<td>0.5762</td>
<td>0.5184</td>
</tr>
<tr>
<td>2</td>
<td>230</td>
<td>15,000</td>
<td>533.54</td>
<td>6.51</td>
<td>0.4724</td>
<td>0.2875</td>
</tr>
</tbody>
</table>

Fig. 15. Pseudocodes for the distribution system planning using the HSG-MOPSO.
SPEA2-MOPSO, and Sigma-MOPSO is carried out with the same performance measurement indicators.

4.3.1. Optimization of PSO parameters

The empirical optimization procedure of the PSO parameters is provided in Fig. 16. The quality of the Pareto-approximation sets obtained with the different values of these parameters is measured using hypervolume and diversity indicators (ref. point: \((2.2 \times 10^6, 1)\)). Some sample results obtained with the 21-node system with different values of population size \(N_p\) and maximum iterations \(I_{\text{max}}\) are given in Table 6. It is observed that the performance of the algorithm is mostly sensitive to these two parameters, i.e., \(N_p\) and \(I_{\text{max}}\). It is also observed that slightly better performance can be obtained with higher values of these two parameters. However, beyond certain point (for example \(N_p=80\) and \(I_{\text{max}}=300\)), the improvement is not significant. Hence, these values are considered as the optimized values for these two parameters. Similarly, the optimized values of the other parameters are obtained and are given in Fig. 16.

4.3.2. Sensitivity tests for the parameter \(K\) and number of strips per objective

The sensitivity tests for the parameter \(K\) and number of strips per objective (of the proposed HSG-MOPSO) are performed for this planning problem. The results of the 54-node system are shown in Fig. 17. The impact of those two parameters on the convergence and diversity are measured by the mean hypervolume and mean diversity indicators, respectively, over 15 independent runs. The results illustrate the convergence is not much sensitive to changes in these parameters. However, the diversity among the solutions is very much sensitive. The reasonable choices for values of these parameters are: \(K=40\) and number of strips per objective=20. These values are used in all subsequent studies for this problem.

4.3.3. Performance assessment

The Pareto-approximation fronts obtained with the HSG-MOPSO, SPEA2-MOPSO, and Sigma-MOPSO of a sample test run are shown in Figs. 18 and 19 for the 21- and 54-node systems, respectively. The performance comparison (over 25 independent test runs for each algorithm) with attainment surface plots is shown in Fig. 20. Table 7 shows a comparison of the means and variances of the hypervolume and diversity indicators.

Remark 1. A better Pareto-approximation front is achieved with the HSG-MOPSO than the other two algorithms. The Pareto-approximation fronts in both problems are highly disconnected. There are some areas with low solution density and the HSG-MOPSO efficiently discovers the solutions. The hypervolume indicators, diversity indicators, and attainment surface plots also illustrate the supremacy of the HSG-MOPSO. It is observed that the hypervolume indicator with the Sigma-MOPSO is better than that of the SPEA2-MOPSO. This is an interesting point contrary to what found in most of the benchmark MOO problems. This illustrates the fact that the Sigma-MOPSO has efficient convergence characteristics, but it is really poor in terms of keeping diversity of the solutions as indicated

---

**Table 6** Sample results of the empirical optimization of PSO parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Performance measuring indicators (mean value)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hypervolume</td>
</tr>
<tr>
<td>(N_p=25) (I_{\text{max}}=100)</td>
<td>0.9661</td>
</tr>
<tr>
<td>(N_p=50) (I_{\text{max}}=100)</td>
<td>0.9521</td>
</tr>
<tr>
<td>(N_p=80) (I_{\text{max}}=100)</td>
<td>0.9062</td>
</tr>
<tr>
<td>(N_p=100) (I_{\text{max}}=100)</td>
<td>0.8889</td>
</tr>
<tr>
<td>(I_{\text{max}}=100) (N_p=80)</td>
<td>0.9062</td>
</tr>
<tr>
<td>(I_{\text{max}}=200) (N_p=80)</td>
<td>0.8497</td>
</tr>
<tr>
<td>(I_{\text{max}}=300) (N_p=80)</td>
<td>0.8019</td>
</tr>
<tr>
<td>(I_{\text{max}}=400) (N_p=80)</td>
<td>0.8088</td>
</tr>
</tbody>
</table>

---

**Fig. 16.** Flowchart of PSO parameter optimization.

**Fig. 17.** Sensitivity tests of: (a) the parameter \(K\) and (b) number of strips per objective (Legends: — hypervolume indicator; ○ diversity indicator).
by the diversity indicator. The attainment surfaces of the HSG-MOPSO and Sigma-MOPSO are very close to each other. In fact, they are found to be incomparable on the 21-node system. However, better attainment is obtained with the HSG-MOPSO on the 54-node system. A small portion of the attainment surfaces is shown in the inset (Fig. 20(b)) to highlight this aspect. The diversity indicator shows a better diversity obtained with the HSG-MOPSO. Overall the performance of the HSG-MOPSO is better both in terms of the convergence and diversity among the solutions.

4.3.4. Analysis of the Pareto-approximation set

The disconnected Pareto-approximation set obtained with the HSG-MOPSO, as shown in Fig. 19 (i.e., for the 54-node system) is taken for further analysis. It is observed that both types of
Sectionalizing switch placement methods yield almost similar Pareto-approximation fronts. It happens so since one method indirectly supports the other method. From the distribution system point of view, a Pareto-approximation set represents a set of radial/meshed distribution networks with varying number of feeders and sectionalizing switches. However, the Pareto-approximation fronts shown in Fig. 19 do not provide any information on the structure of these networks. For this purpose, few bar charts are provided in Figs. 21–23 with the information on the number of loops, sectionalizing switches, and feeders, respectively, for all the non-dominated solutions.

Remark 5. The range of the CLLI for the meshed networks is 0–0.25 and beyond that all the networks are radial. As the number of sectionalizing switches increases, the networks become costlier but more reliable since the CLLI decreases. The networks with

<table>
<thead>
<tr>
<th>Test systems</th>
<th>Algorithms</th>
<th>Hypervolume indicator</th>
<th>Diversity indicator</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean</td>
<td>Variance</td>
</tr>
<tr>
<td>21-node system (ref. point: (2.2 × 10^6,1))</td>
<td>HSG-MOPSO</td>
<td>0.4509</td>
<td>1.032 × 10^{-04}</td>
</tr>
<tr>
<td></td>
<td>SPEA2-MOPSO</td>
<td>0.4375</td>
<td>2.167 × 10^{-04}</td>
</tr>
<tr>
<td></td>
<td>Sigma-MOPSO</td>
<td>0.4385</td>
<td>1.2 × 10^{-03}</td>
</tr>
<tr>
<td>54-node system (ref. point: (3.4 × 10^6,1))</td>
<td>HSG-MOPSO</td>
<td>0.2538</td>
<td>2.45 × 10^{-05}</td>
</tr>
<tr>
<td></td>
<td>SPEA2-MOPSO</td>
<td>0.2509</td>
<td>2.65 × 10^{-06}</td>
</tr>
<tr>
<td></td>
<td>Sigma-MOPSO</td>
<td>0.2546</td>
<td>5.01 × 10^{-06}</td>
</tr>
</tbody>
</table>

Fig. 21. Number of loops in the non-dominated solutions plotted against: (a) objective function-1 and (b) objective function -2.

Fig. 22. Number of sectionalizing switches in the non-dominated solutions plotted against: (a) objective function-1 and (b) objective function -2.
more number of feeders are costlier yet more reliable. This is very much expected from the distribution system point of view. Figs. 22 and 23 show that the density of the non-dominated solutions is higher in the CLLI range of 0–0.5. This happens so because a network with multiple feeders and sectionalizing switches can achieve better CLLI. Thus, very few network structures have the CLLI higher than 0.5.

For the sake of illustration, two sample solution networks, i.e., most reliable and most economical, from the Pareto-approximation set obtained with the HSG-MOPSO are shown in Fig. 24 and the typical features of these networks are given in Table 8.

The distribution system multi-objective planning is a practical problem with disconnected Pareto-optimal fronts. The performance of the HSG-MOPSO on this problem is found to be better than that of the other two algorithms, specifically in terms of diversity of the solutions. Hence, after the experimentation on several problems, i.e., benchmark MOO problems and this.

Table 8

<table>
<thead>
<tr>
<th>Features</th>
<th>Most economical network</th>
<th>Most reliable network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network structure</td>
<td>Radial</td>
<td>Meshed</td>
</tr>
<tr>
<td>Total cost ($)</td>
<td>$4.096 \times 10^6$</td>
<td>$5.1146 \times 10^6$</td>
</tr>
<tr>
<td>CLLI</td>
<td>1</td>
<td>0.0457</td>
</tr>
<tr>
<td>Number of feeders</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Number of sectionalizing switches</td>
<td>0</td>
<td>23</td>
</tr>
<tr>
<td>Number of loops</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>
real-world practical problem, it can be said that the HSG-MOPSO has clear supremacy for the problems with disconnected Pareto-optimal fronts. It has also less computational run-time than that of its immediate competitor, i.e., SPEA2-MOPSO.

5. Conclusion

In this paper, a novel guide selection mechanism is proposed for the multi-objective PSO, named as the HSG-MOPSO, and its performance on several benchmark problems and a real-world problem, i.e., multi-objective planning of electrical power distribution systems, has been investigated. In the HSG-MOPSO, both non-dominated and dominated solutions are considered as potential guides. A guide is assigned to each particle using a novel heuristic judgment. The performance of the proposed algorithm has been compared with two well-known MOPSO approaches, i.e., SPEA2-MOPSO and Sigma-MOPSO, for several benchmark problems. The results show that a very competitive performance can be achieved with the HSG-MOPSO while alleviating some typical problems associated with the other two. Further, in the distribution system planning problem used to illustrate the real-world application of the proposed HSG-MOPSO, a multi-objective planning model has been formulated and solved. A novel feature of this distribution system planning is that both radial and meshed networks are considered. This planning algorithm generates a set of trade-off solutions (using the principle of the Pareto-optimality) representing distribution networks with different structures, i.e., radial/meshed with different number of feeders, and sectionalizing switches. It is noted that the HSG-MOPSO performs very well on all the benchmark problems tested as well as on the distribution systems, has been investigated. In the HSG-MOPSO, both the HSG-MOPSO performs very well on all the benchmark problems tested as well as on the real-world practical problem, it can be said that the HSG-MOPSO has clear supremacy for the problems with disconnected Pareto-optimal fronts. It has also less computational run-time than that of its immediate competitor, i.e., SPEA2-MOPSO.

Appendix

The appendix consists of the list of principal symbols used in this paper and the main computational steps of the SPEA2-MOPSO and Sigma-MOPSO.

A.1. List of principal symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bar{d}$</td>
<td>mean distance among the two neighboring solutions of Pareto-approximation fronts</td>
</tr>
<tr>
<td>$N_{ndf}$</td>
<td>total number of non-dominated solutions</td>
</tr>
<tr>
<td>$C_{IO}$</td>
<td>total installation and operational cost ($)</td>
</tr>
<tr>
<td>$C^b(C^d)$</td>
<td>branch installation cost in $/km (annual branch maintenance cost in $/km/year)</td>
</tr>
<tr>
<td>$l_{ij}$</td>
<td>length of branch between nodes $i, j$</td>
</tr>
<tr>
<td>$A_{w}(E_{fn})$</td>
<td>total number of allowable (existing) branches for feeder routes</td>
</tr>
<tr>
<td>$C^e(t_e)$</td>
<td>cost of energy losses (total planning time in year)</td>
</tr>
<tr>
<td>$y_{ij}$</td>
<td>binary variable (= 1 if the branch $i, j$ is selected as a feeder route, otherwise = 0)</td>
</tr>
<tr>
<td>$C^c$</td>
<td>substation Installation cost in $ (conductor replacement cost in $/km)</td>
</tr>
<tr>
<td>$C_{os} (C_{sw})$</td>
<td>installation cost of one sectionalizing switch (circuit breaker) in $</td>
</tr>
<tr>
<td>$D_f$</td>
<td>discount factor ($D_f = 1/(1+u)^t$; $u$ is interest rate)</td>
</tr>
<tr>
<td>$\eta_{b}(\eta_{a},\eta_{2a})$</td>
<td>total number of branches (substations, sectionalizing switches)</td>
</tr>
<tr>
<td>$\eta_{p}(\eta_{loop})$</td>
<td>total number of feeders (loops)</td>
</tr>
<tr>
<td>$L_{load}(\delta)$</td>
<td>total load (load loss factor)</td>
</tr>
<tr>
<td>$CLI$</td>
<td>contingency-load-loss index for all branch failures taken one at a time</td>
</tr>
<tr>
<td>$N_{DLavg}$</td>
<td>average non-delivered load (non-delivered load due to fault in branch $i$)</td>
</tr>
<tr>
<td>$P^j_{total}(P^j_i)$</td>
<td>total real power loss (real power loss in branch $i$)</td>
</tr>
<tr>
<td>$Z^j(Z^j_i)$</td>
<td>start (end) zone for tie branch of $i$th loop</td>
</tr>
<tr>
<td>$a_{p}(n_{loop}^j)$</td>
<td>total number of load nodes (switches) in feeder $F_i$</td>
</tr>
<tr>
<td>$a$</td>
<td>total number of load nodes served by the substation</td>
</tr>
</tbody>
</table>

A.2. Computational steps for the SPEA2-MOPSO

The main computational steps for the SPEA2-MOPSO are:

Step #1: Start the search with initial randomly chosen position and velocity for all particles.

Step #2: Calculate the objective functions, find out the non-dominated solutions, and store them in an elite archive. Let $Y$ and $Z$ consist solutions of elite archive and PSO population, respectively.

Step #3: Assign fitness to all the members of the PSO population and the elite archive using the SPEA2-based fitness assignment scheme. The fitness assignment steps by the SPEA2 are:

a) Assign strength to all current and archive members. A strength of a member (say $i$) represents number of solutions dominated by it

$$\zeta(i) = \sum_{j \in \{1, 2\}} h_{ij}$$

where $h_{ij} = 1$ if solution $i$ dominates solution $j$ and $h_{ij} = 0$ otherwise; $j \neq i$.

b) Assign a raw fitness $v_{g}(i)$ to member $i$ of the current population as the sum of its dominator’s strength

$$v_{g}(i) = \sum_{j \in \{1, 2\}} q_{ij}\zeta(j)$$

where $q_{ij} = 1$, if solution $j$ dominates solution $i$ and $q_{ij} = 0$, otherwise; $j \neq i$.

c) Measure density around member $i$ of the current population by $i$th nearest neighbor density estimator. The density $\delta(i)$ around member $i$ is defined as
A.3. Computational steps for the Sigma-MOPSO

In the Sigma-MOPSO, the main computational steps are:

Step #1: Starts the search with initial randomly chosen position and velocity for all particles.

Step #2: Calculate the objective functions, find out the non-dominated solutions, and store them in an elite archive.

Step #3: Calculate Sigma values for all the members of the PSO population and elite archive. The Sigma value for a particle with objective functions \( f_1, f_2 \) is calculated as

\[
\sigma = \sqrt{(f_1 - f_2)^2/(f_1^2 + f_2^2)}
\]

Step #4: Update the velocity and position of all the particles using the respective nearest elite solutions with closest Sigma value as guides.

Step #5: Repeat steps 2–4 until a termination criterion is satisfied.

References


