Volume 1, No. 2, February 2013

Journal of Global Research in Mathematical Archives

RESEARCH PAPER

ISSN 2320 – 5822

Available Online at http://www.jgrma.info

TRUST REGION-PARTICLE SWARM FOR MULTI-OBJECTIVE ENGINEERING COMPONENT DESIGN PROBLEMS

Mohamed A. El-Shorbagy[*1], Zeinab M. Hendawy[1], Ahmed A. El-Sawy[2]

[1]Department of Basic Engineering Sciences, Faculty of Engineering, Menofia University, Egypt
Mohammed_shorbagy@yahoo.com[*1] , Zhendawy2010@yahoo.com[1]
[2]Department of Mathematics, Faculty of Science, Qassim University, Saudi Arabia
elsawyus@yahoo.com[2]

Abstract: In this paper, we apply a proposed approach for solving multi-objective engineering design problem (MOEDP) with multiple objectives. In the proposed approach, a reference point based multi-objective optimization (MOO) using a combination between trust region (TR) algorithm and particle swarm optimization (PSO). The integration of TR and PSO has improved the quality of the founded solutions; also it guarantees the faster converge to the Pareto optimal solution. TR has provided the initial set (close to the Pareto set as possible) followed by PSO to improve the quality of the solutions and get all the points on the Pareto frontier. Detailed numerical results on three different MOEDP are reported to demonstrate the effectiveness and advantages of the proposed algorithm for solving practical MOEDP.

Keywords: Multi-objective engineering design problem; trust region; particle swarm optimization;

INTRODUCTION

The MOO is a very important research area in engineering studies because real world design problems require the optimization of a group of objectives. Thanks to the effort of scientists and engineers during the last two decades, particularly the last decade, a wealth of multi-objective (MO) optimizers have been developed, and some multi-objective optimization problems (MOOPs) that could not be solved hitherto were successfully solved by using these optimizers. In terms of robustness and efficiency of the available vector optimizers, these optimizers are still in need of improvements and hence there are many unresolved open problems [1].

This paper intends to present an optimal design of different MOEDPs using hybrid approach which is a combination between numerical optimization method (TR algorithm) and one of the swarm intelligence techniques (PSO optimization). It is a new algorithm that performs TR as deterministic search and PSO as random search.
Trust region is reliable and robust, can be applied to ill-conditioned problems, very strong convergence properties, and has been proven to be theoretically and practically effective and efficient for unconstrained and equality constrained optimization problems [2, 3]. PSO is an evolutionary computational (EC) model and developed by Kennedy and Elberhart [4], which have been inspired by the research of the artificial livings.

In addition, the proposed approach is based on a reference point method which is interactive approach of Wierzbickiis [5], which allows the decision maker to reach solutions close to him important points. For a chosen reference point the closest Pareto optimal solution is the target solution to the reference point method.

There engineering design problems are discussed, two-bar truss design, gear train design, and air-cored solenoid design [6,7]. The results are compared by another approach which solving these design problems to show the reliability of our approach and its ability for solving this kind of problems.

This paper is organized as follows. In section 2, MOO with Reference Point Interactive Approach is described. Section 3 and 4, are provides an overview of the TR algorithm and PSO respectively. The proposed algorithm is presented in section 5. Numerical results are given and discussed in section 6. Finally, section 7 gives a brief conclusion about this study.

MOO WITH REFERENCE POINT INTERACTIVE APPROACH

Definition of MOO

Multiobjective optimization (also called multicriteria optimization, multiperformance or vector optimization) can be defined as the problem of finding a vector of decision variables which satisfies constraints and optimizes a vector function whose elements represent the objective functions. These functions form a mathematical description of performance criteria which are usually in conflict with each other. Hence, the term "optimize" means finding such a solution which would give the values of all the objective functions acceptable to the designer [1,8]. The general minimization problem of q objectives can be mathematically stated as:

$$\left.\begin{array}{lll} \text{minimize :} & f(x) = \left[f_j(x),\ j = 1,2,\ldots,q \right] & \\ \text{subject to the constraints :} & Ci(x) \leq 0, & \mathrm{i} = 1,2,\ldots,p, \\ & Ce(x) = 0, & e = 1,2,\ldots,m, \end{array}\right\} \quad (1)$$

where $f_j(x)$ is the j-th objective function $, Ci(x)$ is the i-th inequality constraint, $C_e(x)$ is the e-th equality constraint and $x = (x_1, x_2, \ldots, x_n)$ is the vector of optimization or decision variables; where n the dimension of the decision variable space. The MOO problem then reduces to finding a x such that $f_j(x)$ is optimized. Since the notion of an optimum solution in MOOP is different compared to the SOOP, the concept of Pareto dominance is used for the evaluation of the solutions. This concept formulated by Vilfredo Pareto is defined as [9]:

Definition 1. (Dominance Criteria [5]). For a problem having more than one objective function (say, f_j, $j = 1,\ldots,q$, $q > 1$), any two solution x_a and x_b can have one of two possibilities, one dominates the other or none dominates the other. A solution x_a is said to dominate the other solution x_b, if both the following condition are true:

- The solution x_a is no worse (say the operator $\prec$ denotes worse and $\succ$ denotes better) than x_b in all objectives, or $f_j(x_a) \not\prec f_j(x_b)$ for all $j = 1,\ldots,q$ objectives.
- The solution x_a is strictly better than x_b in at least one objective, or $f_j(x_a) \succ f_j(x_b)$ for at least one $j \in \{1,\ldots,q\}$.

If any of the above condition is violated, the solution x_a dose not dominates the solution x_b.

Definition 2. (Pareto optimal solution). x^* is said to be a Pareto optimal solution of MOOP if there exists no other feasible x such that, $f_j(x) \leq f_j(x^*)$ for all $j = 1,\ldots,q$ and $f_j(x) < f_j(x^*)$ for at least one objective function f_j.

Reference Point Interactive Approach [10]

As an alternative to the value function methods, Wierzbicki [5] suggested the reference point approach in which the goal is to achieve a weakly, ε-properly or Pareto-optimal solution closest to a supplied reference point of aspiration level based on solving an achievement scalarizing problem. Given a reference point $\bar{z}$ for an M-objective optimization problem of minimizing $(f_1(x), f_2(x), \ldots, f_q(x))$ with x belongs to the search space, the following single-objective optimization problem is solved for this purpose:

$$\left.\begin{array}{lll} \text{minimize :} & \max_{j=1}^{q} \left[w_j \left(f_j(x) - \bar{z}_j \right) \right] & \\ \text{subject to:} & Ci(\mathrm{x}) \leq 0, & \mathrm{i} = 1,2,\ldots,l, \\ & Ce(x) = 0, & e = 1,2,\ldots,m, \end{array}\right\} \quad (2)$$

Here, w_j is the j-th component of a chosen weight vector used for scalarizing the objectives. Figure 1 illustrates the concept. For a chosen reference point, the closest Pareto optimal solution (in the sense of the weighted-sum of the objectives) is the target solution to the reference point method. To make the procedure interactive and useful in practice, Wierzbicki [5] suggested a procedure in which the obtained solution z' is used to create q new reference points, as $z_i = \bar{z} + (z' - \bar{z}) \cdot e_i$;

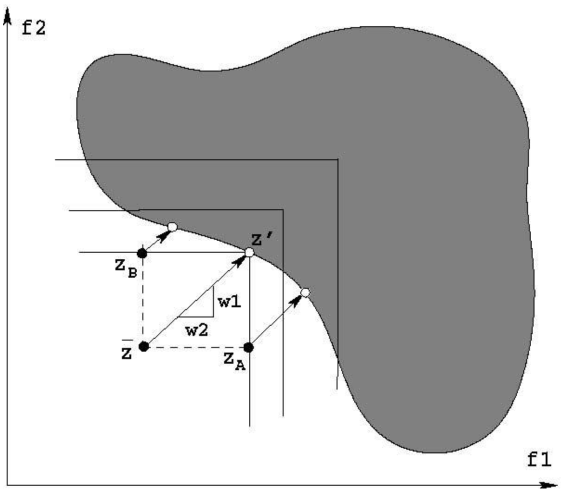

Figure 1. Classical reference point approach.

where e_i is the i-th coordinate direction vector. For the two-objective problem shown in the figure, two such new reference points (z_A and z_B) are also shown. New Pareto-optimal solutions are then found by forming new achievement scalarizing problems. If the decision-maker is not satisfied with any of these Pareto-optimal solutions, a new reference point is suggested and the above procedure is repeated. It is interesting to note that the reference point may be a feasible one (deducible from a solution vector) or an infeasible point which cannot be obtained from any solution from the feasible search space. If a reference point is feasible and is not a Pareto-optimal solution, the decision-maker may then be interested in knowing solutions which are Pareto-optimal and close to the reference point. On the other hand, if the reference point is an infeasible one, the decision-maker would be interested in finding

Pareto-optimal solutions which are close to the supplied reference point.
To utilize the reference point approach in practice, the decision-maker needs to supply a reference point and a weight vector at a time. The location of the reference point causes the procedure to focus on a certain region in the Pareto-optimal frontier, whereas a supplied weight vector makes a finer trade-off among the objectives and focuses the procedure to find a single Pareto-optimal solution (in most situations) trading-off the objectives. Thus, the reference point provides a higher-level information about the region to focus and weight vector provides a more detailed information about what point on the Pareto-optimal front to converge.

TRUST REGION ALGORITHM

Trust region method generate steps with the help of a quadratic model of the objective function, define a region around the current iterate within which they trust the model to be an adequate representation of the objective function, and then choose the step to be approximate minimzer of the model in this region. If a step is not acceptable, they reduce the size of the region and find a new minimize. In general, the direction of the step changes whenever the size of the TR is altered [11,12]. To see the idea of TR, consider the unconstrained optimization problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\ f(x) \tag{3}$$

where $f(x)$ is a nonlinear continuous differentiable function in $\mathbb{R}^n$. For a known iterate x_k the TR method determines subsequent iterate using

$$x_{k+1} = x_k + d_k, \tag{4}$$

where d_k is trial step determined by minimizing a local quadratic (approximating) model of f at x_k (TR sub-problem) given by

$$\begin{aligned} &\text{minimize} \quad q_k(d) = f_k + \nabla f_k^T d + \frac{1}{2} d^T H_k d \\ &\text{subject to} \quad \|d\| \le \Delta_k, \end{aligned} \tag{5}$$

where H_k is hessian of $f(x)$ or approximate to it, and $\Delta_k > 0$ is the TR radius. Using the ratio

$$r_k = \frac{f(x_k) - f(x_k + d_k)}{q_k(0) - q_k(d_k)}, \tag{6}$$

traditional TR methods evaluate an agreement between the model and the objective function. The trial step d_k is accepted whenever r_k is greater than a positive constant. This leads us to the new point $x_{k+1} = x_k + d_k$, and the TR radius is updated. Otherwise, the TR radius must be diminished and the sub-problem (5) must be solved again [13].
Because of the boundedness of the TR, TR algorithms can use non-convex approximate models. This is one of the advantages of TR algorithms comparing with line search algorithms. TR algorithms are reliable and robust, they can be applied to ill-conditioned problems, they have very strong convergence properties, and have been proven to be theoretically and practically effective and efficient for unconstrained and equality constrained optimization problems [2,14,15]. Also, The TR algorithm has proven to be a very successful globalization technique for nonlinear programming problems with equality and inequality constraints [3,16,17].
For MOOPs, Kim and Ryu [18] developed an iterative algorithm for bi-objective stochastic optimization problems based on the TR method and investigated different sampling schemes. Their algorithm does not require any strong modeling assumptions, and has great potential to work well in various real-world settings. El-Sobky [19] used the TR algorithm in solving an interactive approach for MOOPs; where an active set strategy is used together with a reduced Hessian technique to convert the single objective optimization problem with quality and inequality constraints to equality constrained optimization problem and the computation of the trial step to two easy TR sub-problems similar to those for the unconstrained case.

PARTICLE SWARM OPTIMIZATION

PSO is an EC model which is based on swarm intelligence. PSO is developed by Kennedy and Elberhart [4] who have been inspired by the research of the artificial livings. Similar to EC techniques, PSO is also an optimizer based on population. The system is initialized firstly in a set of randomly generated potential solutions, and then performs the search for the optimum one iteratively. Whereas the PSO does not possess the crossover and mutation processes used in EC, it finds the optimum solution by swarms following the best particle. Compared to EC, the PSO has much more profound intelligent background and could be performed more easily. Based on its advantages, the PSO is not only suitable for science research, but also engineering applications, in the fields of evolutionary computing, optimization and many others.

The basic PSO algorithm is constructed as follows: Consider a swarm of N particles or birds. For particle i, it was originally proposed that the position x_i is updated in the following manner:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{7}$$

with the velocity v_i^{t+1} calculated as follows:

$$v_i^{t+1} = w v_i^t + c_1 r_1 (p_i - x_i^t) + c_2 r_2 (p_g - x_i^t) . \tag{8}$$

Here, subscript t indicates an pseudo-time increment. p_i represents the best ever position of particle i at time t, with p_g representing the global best ever position in the swarm at time t . r_1 and r_2 represent uniform random numbers between 0 and 1. Figure 2 shows the Description of velocity and position updates of a particle for a two-dimensional parameter space.

MOO has been one of the most studied application areas of PSO algorithms. Number of approaches have been utilized and/or designed to tackle MOOPs using PSO. A comprehensive survey of the state-of-the-art in Multi-objective particle swarm optimizers can be found in [20] where different techniques reported in multi-objective PSO development have been categorized and discussed.

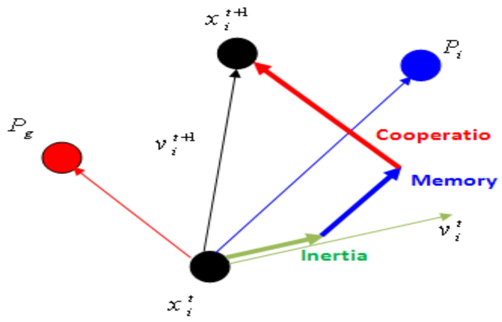

Figure 2. Description of velocity and position updates in PSO for a two dimensional parameter space.

THE PROPOSED APPROACH

In this section, the proposed algorithm is presented. The proposed algorithm contains three stages initialization stage, TR stage and PSO stage.

Initialization stage

1- Initialization

Initialize N reference points in the search space, TR parameters, and PSO parameters.

2- Reference Point method

The classical reference point approach discussed above, will find a solution depending on the chosen weight vector and is therefore subjective. Moreover, the single solution is specific to the chosen weight vector and does not provide any information about how the solution would change with a slight change in the weight vector. To find a solution for another weight vector, a new achievement scalarizing problem needs to be formed again and solved. Moreover, despite some modifications [21], the reference point approach works with only one reference point at a time. However, the decision-maker may be interested in exploring the preferred regions of Pareto-optimality for multiple reference points simultaneously. With the above principles of reference point approaches and difficulties with the classical methods, we use a methodology by which a set of Pareto-optimal solutions near a supplied set of reference points will be found, thereby eliminating the need of any weight vector and the need of applying the methodologies again and again.

Given a reference point for an q-objective optimization problem of minimizing $(f_1(x),...,f_q(x))$ with x belongs to the search space, the following single-objective optimization problem is solved:

$$\left.\begin{aligned} &\text{minimize}: \; f(x) = \left(\sum_{j=1}^{q} \left|f_j(x) - \bar{z}_j\right|^p\right)^{1/p} \\ &\text{subject to:} \; Ci(x) \le 0, \quad \mathrm{i} = 1,2,...,l, \\ &\qquad\qquad Ce(x) = 0, \quad e = 1,2,...,m, \end{aligned}\right\} \tag{9}$$

where the parameter p can take any value between 1 and ∞. When $p = 2$ is used, an Euclidean distance of any point in the objective space from the reference point $\bar{z}$ is minimized.

TR Stage

This section is devoted to presenting the detailed description of TR algorithm for solving problem (9) (see [22]). The TR algorithm combines ideas from Byrd [23], Omojokun [24], El Alem [25].

Following Dennis et al. [26], we define the indicator matrix $W(x) \in \mathbb{R}^{p \times p}$, whose diagonal entries are

$$w_i(x) = \begin{cases} 1 & \text{if } Ci(x) \geq 0, \\ 0 & \text{if } Ci(x) < 0. \end{cases} \tag{10}$$

Using this matrix, the Problem defined in Eq. (9) can be transformed to the following equality constrained optimization problem:

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && 1/2Ci(x)^T W(x) Ci(x) = 0, \\ &&& Ce(x) = 0. \end{aligned} \tag{11}$$

The above problem can be rewritten as:

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && h(x) = 0, \end{aligned} \tag{12}$$

where $h(x) = [Ce(x) \quad 1/2Ci(x)^T W(x) Ci(x) = 0]$.

The Lagrangian function associated with problem defined in (12) is given by

$$L(x_k, \lambda_k) = f(x_k) + \lambda_k^T h(x_k) \tag{13}$$

where $\lambda_k \in \mathbb{R}$ is the Lagrange multiplier vector associated with equality constraint $h(x_k) \in \mathbb{R}$.

The augmented Lagrangian is the function

$$\Phi(x, \lambda; r) = L(x, \lambda) + r\|h(x_k)\|^2, \tag{14}$$

where $r > 0$ is a parameter usually called the penalty parameter.

The reduced Hessian approach is used to compute a trial step d_k. In this approach, the trial step d_k is decomposed into two orthogonal components; the normal component d_k^n and the tangential component d_k^t. The trial step d_k has the form $d_k = d_k^n + Z_k \bar{d}_k^t$, where Z_k is a matrix whose columns form an orthonormal basis for the null space of $\nabla h(x_k)^T$.

We obtain the normal component d_k^n by solving the following TR sub-problem:

$$\begin{aligned} &\text{minimize} && \frac{1}{2}\left\|h(x_k) + \nabla h(x_k)^T d^n\right\|^2 \\ &\text{subject to} && \|d^n\| \leq \xi \Delta_k, \end{aligned} \tag{15}$$

for some $\xi \in (0,1)$.

Given the normal component d_k^n, we compute the tangential component $d_k^t = Z_k \bar{d}_k^t$ by solving the following TR sub-problem:

$$\begin{aligned} &\text{minimize} && \left[Z_k^T\left(\nabla_x L(x_k, \lambda_k) + H_k d_k^n\right)\right]^T \bar{d}^t + \frac{1}{2}\bar{d}^{t^T} Z_k^T H_k Z_k \bar{d}^t \\ &\text{subject to} && \left\|Z_k \bar{d}^t\right\| \leq \sqrt{\Delta_k^2 - \left\|d_k^n\right\|^2}, \end{aligned} \tag{16}$$

Once the trial step is computed, it needs to be tested to determine whether it will be accepted or not. To do that, a merit function is needed. We use the augmented Lagrangian function (14) as a merit function. To test the step, we compare the actual reduction in the merit function in moving from x_k to $x_k + d_k$ versus the predicted reduction.

The actual reduction in the merit function is defined as:

$$Ared_k = L(x_k, \lambda_k) - L(x_{k+1}, \lambda_{k+1}) + r_k\left[\|h(x_k)\|^2 - \|h(x_{k+1})\|^2\right],. \tag{17}$$

The predicted reduction in the merit function is defined as:

$$Pred_k = -\nabla_x L(x_k, \lambda_k)^T d_k - \frac{1}{2} d_k^T H_k d_k - \Delta\lambda_k^T\left(h(x_k) + \nabla h(x_k)^T d_k\right) + r_k\left[\|h(x_k)\|^2 - \left\|h(x_k) + \nabla h(x_k)^T d_k\right\|^2\right]; \tag{18}$$

where $\Delta\lambda_k = (\lambda_{k+1} - \lambda_k)$.

If $(Ared_k / Pred_k) < \tau_0$, where $\tau_0 \in (0,1)$ is a small fixed constant, then the step is rejected. In this case, the radius of the TR Δ_k is decreased by setting $\Delta_k = \tau_3 \|d_k\|$, where $\tau_3 \in (0,1)$, and another trial step is computed using the new trust-region radius. If $(Ared_k / Pred_k) \geq \tau_2$, where $\tau_2 > 0$, then the step is accepted and set the TR as $\Delta_{k+1} = \min(\Delta_{\max}, \max(\Delta_{\min}, \tau_1 \Delta_k))$. If $\tau_0 \leq (Ared_k / Pred_k) < \tau_2$, then the step is accepted and set the TR as $\Delta_{k+1} = \max(\Delta_k, \Delta_{\min})$. Finally, the algorithm is terminated when either $\|d_k\| \leq \varepsilon_1$ or $\left\|Z_k^T \nabla_x L_k\right\| + \|h_k\| \leq \varepsilon_2$, for some $\varepsilon_1, \varepsilon_2 > 0$. The pseudo code of TR stage showing in Fig. 3.

Choose ε_1, ε_2, τ_0, τ_1, τ_2, τ_3, $\Delta_0, \Delta_{\max}, \Delta_{\min}$ such that $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $0 < \tau_3 < 1 < \tau_1$, $0 \leq \tau_0 \leq \tau_2 < 1$, $\tau_2 > 0$, and $\Delta_{\min} \leq \Delta_0 \leq \Delta_{\max}$

For each point N $x_0 \in \mathbb{R}^n$, compute W_0 , $H_0 \in \mathbb{R}^{n\times n}$, and set $k = 0$.
If $\left\| Z_k^T \nabla_x L_k \right\| + \left\| h_k \right\| \le \varepsilon_2 \rightarrow$ end for
Solve the sub-problem (15) to give the normal component d_k^n
Solve the sub-problem (16) to give the tangential component $d_k^t = Z_k \bar{d}_k^t$
Compute the trial step $d_k = d_k^n + Z_k \bar{d}_k^t$
If $\left\| d_k \right\| \le \varepsilon_2 \rightarrow$ end for
Compute $Ared_k$ and $Pred_k$
While $Ared_k / Pred_k < \tau_0 \rightarrow \Delta_k = \tau_3 \left\| d_k \right\| \rightarrow$ compute a new trial step d_k
If $\tau_0 \le Ared_k / Pred_k < \tau_2$, then $x_{k+1} = x_k + d_k \rightarrow \Delta_{k+1} = \max\ \Delta_k, \Delta_{\min}$.
Else $Ared_k / Pred_k \ge \tau_2$, then $x_{k+1} = x_k + d_k \rightarrow \Delta_{k+1} = \min\ \Delta_{\max},\ \max\ \Delta_{\min}, \tau_1 \Delta_k$.
Update H_{k+1}, W_{k+1} , and set $k = k + 1$

Figure 3. The pseudo code of TR stage

PSO stage

In this stage a homogeneous PSO for MOOP (see [27]) is proposed with a decreasing constriction factor to restrict velocity of the particles and control it [28]. In homogeneous PSO one global repository concept is proposed for choosing *pbest* and *gbest*, this means that each particle has lost its own identity and treated simply as a member of social group. The procedure of the PSO stage is as follows.

Step 1: Initialization

All non-dominated points (which obtained by applying TR stage) chosen as particles position x_i^t .

Store non-dominated particles in Pareto repository. If the specific constraint doesn't exist for a repository, the size of the repository is unlimited.

Step 2: Evaluation

Evaluate the MO fitness value of each particle and save it in a vector form.

Step 3: Floating

Two optimal solutions are chosen randomly for *pbest* and *gbest* from the repository.

Determine the new position of each particle with Eqs. (7) and (8).

Step 4: Repairing of particles:

Where the particle i starts at position x_i^t with velocity v_i^t in the feasible space, the new position x_i^{t+1} depends on velocity v_i^{t+1}, so we introduce a modified constriction factor (i.e., decreasing constriction factor)

$$\chi = \frac{2}{\left| -2 - \tau - \sqrt{\tau^2 + \tau} \right|}; \tag{19}$$

where, τ is the age of the infeasible particle (i.e., how long it is still infeasible) and it is increased with the number of failed trials to keep the feasibility of the particle.

The new modified positions of the particles are computed as:

$$x_i^{t+1} = x_i^t + \chi\, v_i^{t+1}. \tag{20}$$

Step 5: Selection and update the repository

Check the Pareto optimality of each particle. If the fitness value of the particle is non-dominated when it compared to the Pareto optimal set in a repository, save it into the Pareto repository.

In the Pareto repository, if a particle is dominated from new one, then discard it.

Step 6: Repeat

Repeat again step 2 to step 5 until the number of generation reaches to given t . The pseudo code of PSO stage showing in Fig. 4.

Store non-dominated solution in Pareto repository
Chose non-dominated solution as position of particles x_i^t .
Initialize parameters for PSO v_i^t, w , c_1, c_2 .
While (number of iterations, or the stopping criterion is not met)
Chosen randomly pbest and gbest from the repository.
Update particles velocity v_i^{t+1} and position x_i^{t+1} according to equation (7) and equation (8) of all particles.
Repair the unfeasible particle according to equation (20).
Evaluate fitness of particle swarm
Selection and update the repository
End while

Figure 4. The pseudo code of PSO stage.

ENGINEERING COMPONENT DESIGN PROBLEMS

In the following, we discuss three engineering component design problems, two-bar truss design, gear train design, and air-cored solenoid design [6,7]. We have kept the proposed approach parameters same in all problems as is shown in Table I (see [28,29,30]). The algorithm is coded in MATLAB 7.2 and the simulations are run on a Pentium 4 CPU 900 MHz with 512 MB memory capacity.

Table I. The Parameter Adopted In The Implementation Of The Proposed Algorithm

Parameter	***Value***	***Parameter***	***Value***
N	20-50	Δ_{max}	105 Δ_0
ε_1, ε_2	10-7	Δ_{min}	10-3
τ_0	0	PSO iteration	300
τ_1	2	w	0.6
τ_2	0.25	c_1	2.8
τ_3	0.25	c_2	1.3
Δ_0	1,1.5 $\times \Delta_{min}$	τ	15

Two-bar truss design

This problem was originally studied using NSGA-II [6]. The truss (Fig. 5) has to carry a certain load without elastic failure. Thus, in addition to the objective of designing the truss for minimum volume (which is equivalent to designing for minimum cost of fabrication), there are additional objectives of minimizing stresses in each of the two members AC and BC. We construct the following two-objective optimization problem for three variables y (vertical distance between B and C in m), x1 (length of AC in m) and x2 (length of BC in m):

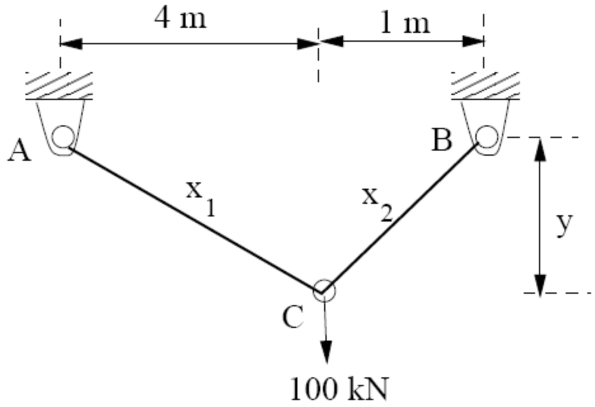

Figure 5. The two-bar truss is shown.

$$
\begin{aligned}
&\text{Min} && f_1 = x_1\sqrt{16+y^2} + x_2\sqrt{1+y^2} \\
&\text{Min} && f_2 = \max\ \sigma_{AC}, \sigma_{BC} \\
&\text{subject to} && \max\ \sigma_{AC}, \sigma_{BC} \le 1\ 10^5 \\
& && 1 \le y \le 3 \quad \text{and} \quad 0 \le x_1, x_2 \le 0.01
\end{aligned}
$$

The stresses are calculated as follows:

$$
\sigma_{AC} = \frac{20\sqrt{16+y^2}}{yx_1} \qquad \sigma_{BC} = \frac{80\sqrt{1+y^2}}{yx_2}
$$

Figure 6 shows the optimized front found using the proposed method (476 points) and NSGA-II. The solutions are spread by NSGA-II in the following range: (0.00407 m3, 99755 kPa) and (0.05304 m3, 8439 kPa), while by the proposed approach :(0.0040547 m3, 99599 kPa) and (0.081623 m3, 8432.74 kPa), which indicates the power of proposed approach compared to NSGA-II.

What is also important that all these solutions have been found in just one simulation run of our approach. From the figure below, we can see that our approach solutions are better than NSGA-II solutions, both in terms of closeness to the optimum front and in their spread.

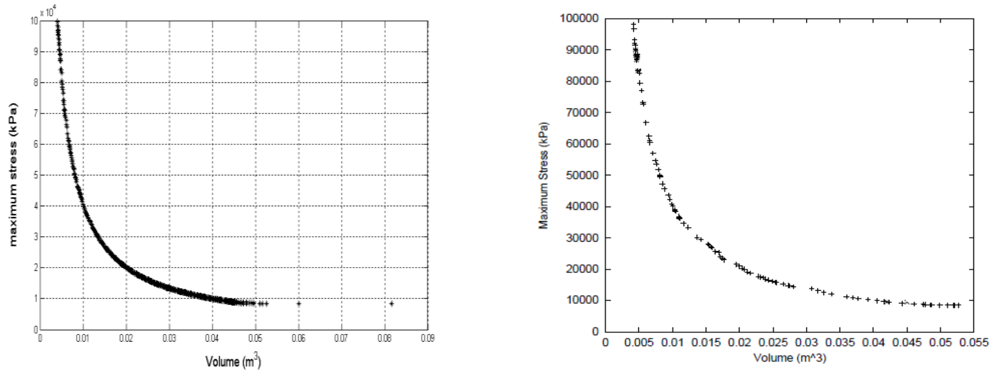

Figure 6. Optimized solutions obtained using the proposed approach (left) and NSGA (right) for the two-bar truss problem.

If minimization of stress is important, the proposed approach finds a solution with stress as low as 8432.74 kPa, where as NSGA-II method has found a solution with minimum stress of 8439 kPa. On the other hand, If minimization of volume is important, the proposed approach finds a solution with volume as low as 0.0040547 m3, where as NSGA-II method has found a solution with minimum stress of 0.00407 m3. The following Table shows the best maximum stress and best volume obtained by proposed algorithm.

Table II. The Best Maximum Stress And Best Volume Obtained By The Proposed Algorithm

	x_1	x_2	y	f_1 Volume	f_2 Maximum stress
Min. Maximum stress	0.01	0.01	3	0.081623	8432.74
Min. Volume	0.00044177	0.00091105	2.0413	0.0040547	99599

Gear train design

A compound gear train is to be designed to achieve a specific gear ratio between the driver and driven shafts (Fig. 7).
The objective of the gear train design is to find the number of teeth in each of the four gears so as to minimize (i) the error between the obtained gear ratio and a required gear ratio of 1/6.931 and (ii) the maximum size of any of the four gears. Since the number of teeth must be integers, all four variables are strictly integers. By denoting the variable vector x=(x1; x2; x3; x4)= (Td; Tb; Ta; Tf), we write the two-objective optimization problem:

$$\begin{array}{ll} \text{Min} & f_1 = \left[\dfrac{1}{6.931} - \dfrac{x_1 x_2}{x_3 x_4} \right]^2 \\ \text{Min} & f_2 = \max\ x_1, x_2, x_3, x_4 \\ \text{subject to} & 12 \le x_1, x_2, x_3, x_4 \le 60 \\ & \text{all } x_i\text{'s are integers.} \end{array}$$

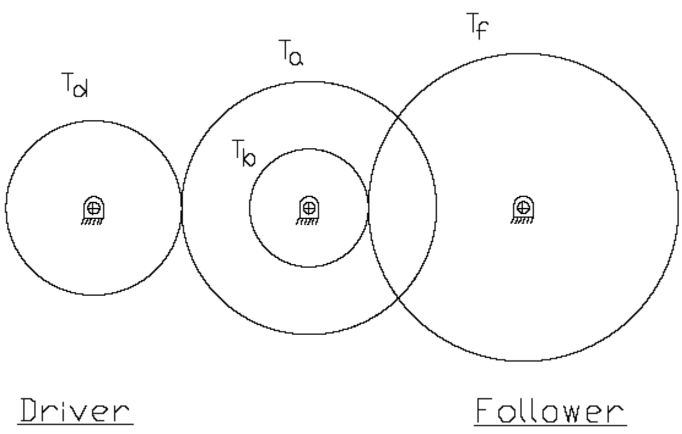

Figure 7. A compound gear train is shown.

Figure 8 shows the obtained optimized solutions by our approach (22 point) and NSGA-II [6], while Fig. 9 shows the non-dominated front obtained using the proposed approach. Table 3 shows the best error and the best maximum size obtained by proposed algorithm as compared to NSGA-II. It can be deduced that the proposed algorithm finds comparable minimum of maximum size to NSGA-II.

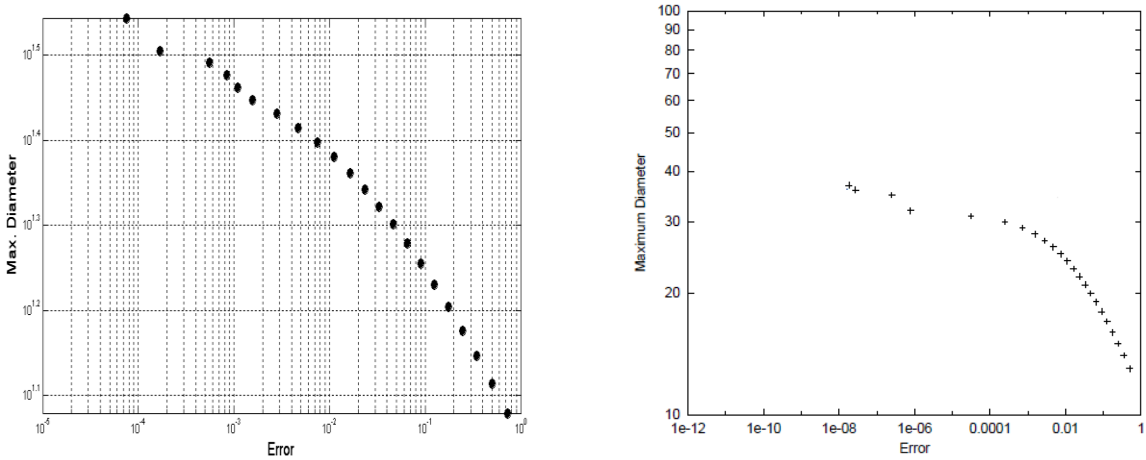

Figure 8. Optimized solutions obtained using the proposed approach (left) and NSGA (right) for the gear train design problem.

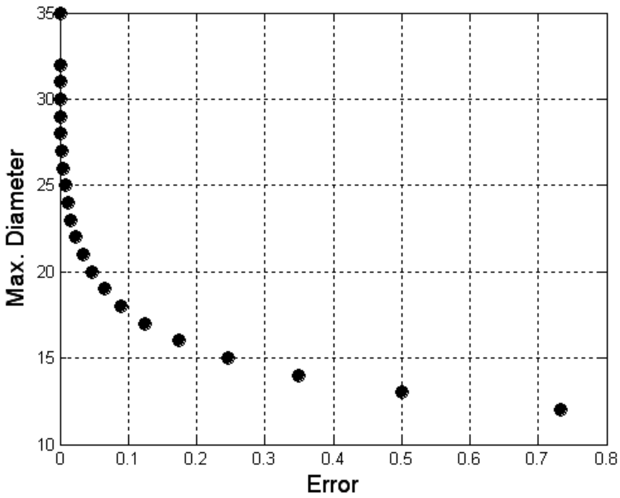

Figure 9. Non-dominated front obtained by the proposed approach for the gear train design problem

Table III. The Best Error And The Best Maximum Size Obtained By Proposed Algorithm And NSGA-II

	NSGA-II	***Proposed approach***		***NSGA-II***	***Proposed approach***
Best Error	1.83 (10^{-8})	7.5028e-005	***Corresponding Error***	2.47 (10^{-4})	**0.732**26
Corresponding Max. Diameter	37	35	***Best Max. Diameter***	30	12
x_1	12	14	x_1	12	12
x_2	12	13	x_2	12	12
x_3	27	35	x_3	30	12
x_4	37	34	x_4	30	12

Shape design of an air-cored solenoid

The multiobjective shape optimization of a coreless solenoid with rectangular cross section a × b and a mean radius c (see Fig. 10) is tackled [7]. If the electric current is uniformly distributed over the cross section, it can be shown that if the number of turns (N) of the solenoid is given, the inductance L[μH] can be approximated from

$$L = \frac{31.49\ \ a^2N^2/b}{9+6\ \ a/b\ \ +10\ \ a/b}$$

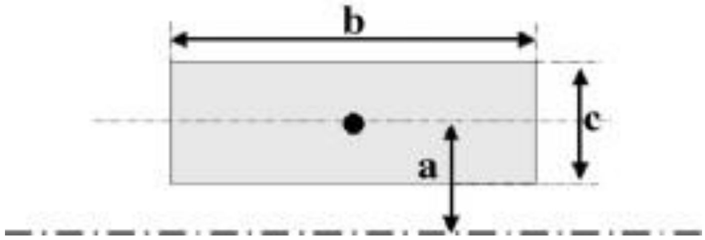

Figure 10. Cross section of the solenoid and design variables

This multiobjective design problem can then be formally defined in the following two terms: maximize the inductance L(a, b, c) and minimize the volume V(a, b, c) for the given length $k_1 = 10$ m and $k_2 = 10^{-6}$ m2 of the current carrying wire. In order to simplify the analysis, two variables, a and b, are considered. Correspondingly, the computation of L and V are simplified, respectively, to

$$f_1 = \frac{31.49\ \ k_1^2/4\pi^2 b}{9+6\ \ a/b\ \ +5\ \ k_1k_2/\pi ab^2}$$

$$f_2 = \frac{\pi a^2 b}{4} + \frac{k_1^2 k_2^2}{4\pi a^2 b} + \frac{k_1 k_2}{2}$$

Now, the problem reads: maximize $f_1\ \ a,b$ and minimize $f_2\ \ a,b$ subject to

$$a > \sqrt{\frac{k_1 k_2}{4\pi b}}, \quad a \in\ \ 0,0.1\ , \quad b \in\ \ 0,0.3\ .$$

Despite the simplicity of formulas for both objective functions, the multiobjective optimization problem is not trivial and cannot be tackled analytically. The searched Pareto front using the proposed algorithm and using [7] are illustrated in Fig. 11. Clearly, the proposed algorithm produces a better uniform sampling of the Pareto front for this application (928 point) than it obtained by [7]. The following Table shows the best volume and best Inductance obtained by proposed algorithm.

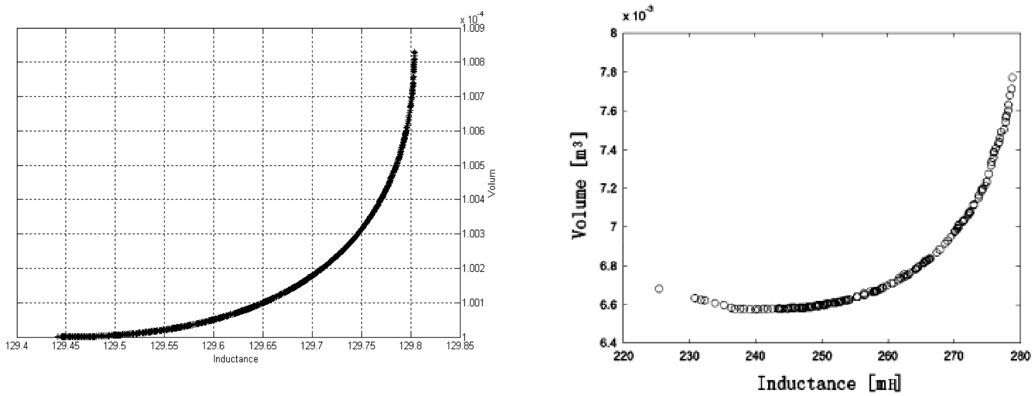

Figure 11. Optimized solutions obtained using the proposed approach (left) and [7] (right) for the Shape design of an air-cored solenoid

Table IV. The Best Volume And Best Inductance Obtained By Proposed Algorithm

Solution	*a*	*b*	f_1 Inductance	f_2 Volume
Best Volume	0.037544	0.022582	129.441486	0.0001
Best Inductance	0.034158	0.022741	129.803667	0.00010083

The table shows that a wide variety of optimal solutions have been obtained. Solutions in [7] are not as good as our approach solutions.

CONCLUSION

In this paper, we have used a new approach for finding multiple Pareto-optimal solutions in a number of engineering design problems. In the proposed approach, a reference point based MOO using a Hybrid between TR algorithm and PSO. In this approach, we introduced an integration between TR and PSO to improve the quality of the founded solutions, and also to ensure faster convergence to the Pareto optimal solution. TR has provided the initial set (close to the Pareto set as possible and the reference point of the DM) followed by PSO to improve the quality of the solutions and get all the points on the Pareto frontier.
The results on three engineering design problems show that a wide spread of solutions have been obtained. In all problems, the proposed approach finds a front better and wider than that found by other approaches. The study are encouraging and suggests immediate application of the proposed method to more complex engineering problems.

REFERENCES

[1] C. C. Coello, “An Updated Survey of GA-Based Multiobjective Optimization Techniques”, ACM Computing Surveys, ACM Press, Vol. 32, No. 2, pp. 109--143, June 2000.

[2] M. Ahookhosh, K. Amini, "A Nonmonotone trust region method with adaptive radius for unconstrained optimization problems", Computers & Mathematics with Applications 60(3): 411-422 (2010).

[3] B. El-Sobky, A Global Convergence Theory for Trust-Region Algorithm for General Nonlinear Programming Problem, Publicities in Fourth Saudi Science Conference, Al-Madinah Al-Munawarah, Kingdom of Saudi Arabia, March, 21st -24th, 2010.

[4] J. Kennedy, R. C. Eberhart, and Y. Shi, "Swarm Intelligence", Morgan Kaufmann, 2001.

[5] K. Miettinen, Nonlinear Multiobjective Optimization, Kluwer Academic Publishers, Boston, 1999.

[6] K. Deb, A. Pratap, S. Moitra " Mechanical Component Design for multi-objective using Elitist non-dominated sorting GA." KanGAL Report No. 200002, 2000.

[7] Y. Wang, S. Yang, G. Ni, S.L. Ho, Z.J. Liu, “An emigration genetic algorithm and its application to multiobjective optimal designs of electromagnetic devices”, IEEE TRANSACTIONS ON MAGNETICS, VOL. 40, NO. 2, MARCH 2004.

[8] A. Osyczka, “Multicriteria optimization for engineering design”, In John S. Gero, editor, Design Optimization, pages 193–227. Academic Press, 1985.

[9] V. Pareto, “Cours d’Économie politique”, volume I and II. F. Rouge, Lausanne, 1896-97.

[10] K. Deb, J. Sundar, Udaya Bhaskara Rao N. and Shamik Chaudhuri, Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms, International Journal of Computational Intelligence Research, Vol.2, No.3 (2006), pp. 273–286.

[11] X. W. Liu and Y. X. Yuan, "A robust trust region algorithm for solving general nonlinear programming", J. Comput. Math., 19: 309-322 (2001).

[12] F. Wang, K. Zhang, C. Wang, L. Wang, A variant of trust-region methods for unconstrained optimization, Applied Mathematics and Computation 203(1): 297-307 (2008).

[13] M. Ahookhosh, K. Amini, M. R. Peyghami, "A nonmonotone trust-region line search method for large-scale unconstrained optimization", Applied Mathematical Modelling, 36: 478–487 (2012).

[14] Z. Shi, J. Guo, "A new trust region method for unconstrained optimization", Journal of Computational and Applied Mathematics 213: 509 – 520 (2008).

[15] J. Zhang, K. Zhang, S. Qu, "A nonmonotone adaptive trust region method for unconstrained optimization based on conic model", Applied Mathematics and Computation 217(8): 4265-4273 (2010).

[16] B. El-Sobky, "A global convergence theory for an active trust region algorithm for solving the general nonlinear programming problem", Applied Mathematics and computation archive,144 (1):127-157 (2003).

[17] Y. Ji, K. Zhang, S. Qu, Y. Zhou, "A trust-region method by active-set strategy for general nonlinear optimization", Computers and Mathematics with Applications 54: 229–241 (2007).

[18] S. Kim, J. Ryu, "A Trust-Region Algorithm for Bi-Objective Stochastic Optimization", Procedia Computer Science, 4: 1422-1430 (2011).

[19] B. El-Sobky, "An Active-Set Trust-Region Algorithm for Solving Constrained Multi-Objective Optimization Problem", Applied Mathematical Sciences, 6: 1599 – 1612 (2012).

[20] M. R. Sierra, and C. C. Coello, “Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the- Art”, International Journal of Computational Intelligence Research, 2(3): 287-308, 2006.

[21] A. M. J. Skulimowski, “Classification and properties of dominating points in vector optimization”, In P. Klein-schmidt, F. J. Radermacher, W. Scweitzer, and H. Wildermann, editors, Methods of Operations Research 58, pages 99–112. Frankfurt am Main, Germany: Athenum Verlag, 1989.

[22] [22] A. A. El-Sawy, Z. M. Hendawy, M. A. El-Shorbagy, Combining Trust-Region Algorithm and local search for Multi-objective Optimization, 3rd European Conference of Civil Engineering (ECCIE'12), Paris, France, December, 11-12, 2012.

[23] R. Byrd, "Robust Trust Region methods for Nonlinearly Constrained Optimization", A talk presented at the second SIAM conference on optimization, Houston, TX (1987).

[24] E. Omojokun, "Trust-region strategies for optimization with nonlinear equality and inequality constraints", PhD thesis, Department of Computer Science, University of Colorado, Boulder, Colorado, 1989. ditor, Spares Matrices and Thier Uses, pages 57-87. Academic Press, New York.

[25] M. El-Alem, "A robust trust-region algorithm with a non-monotonic penalty parameter scheme for constrained optimization", SIAM J. Optim. 5 (2): 348-378 (1995).

[26] J. Dennis, M. El-Alem, and K. Williamson, "A trust-region approach to nonlinear systems of equalities and inequalities", SIAM J Optimization, 9: 291-315 (1999).

[27] S. K. Hwang, K. Koo, and J. S. Lee, "Homogeneous Particle Swarm Optimizer for Multi-objective Optimization Problem", ICGST International Journal on Artificial Intelligence and Machine Learning, AIML, 2006.

[28] W. F. Abd-El-Wahed, A. A. Mousa, M. A. El-Shorbagy, "Integrating particle swarm optimization with genetic algorithms for solving nonlinear optimization problems", J. Computational Applied Mathematics 235(5): 1446-1453 (2011).

[29] I. S. Duff, J. Nocedal, and J.K. Reid, ”The use of linear programming for the solution of sparse sets of nonlinear equations”, SIAM J. Sci. Stat. Comput. 8: 99-108 (1987).

[30] R. Fletcher, "Practical Methods of Optimization", (second edition) (JohnWiley and Sons, Chichester, 1987).

SHORT BIODATA OF ALL THE CORESPONDING AUTHOR

Mohamed A. El-Shorbagy was born in Ashmoon, Menofia, Egypt, on April 3, 1982. He received the B.Sc. degree in Electrical Engineering and the M.Sc. degree in engineering mathematics from Faculty of Engineering, Menofia University, Shebin El-Kom, Egypt, in 2004 and 2010, respectively. He is working towards the Ph.D. degree in the Department of Basic Engineering Sciences at the same college.

He is currently an Assistant Lecture in Department of Basic Engineering Sciences, Faculty of Engineering, Menofia University, Shebin El-Kom, Egypt. He is a member of the World Scientific and Engineering Academy and Society. His current research interests include: evolutionary multiobjective optimization, Swarm Intelligence, Numerical optimization and evolutionary algorithms in general.

