Particle swarm optimization technique with stochastic ranking for constrained optimization problems

Layak Ali, Samrat L. Sabat
School of Physics, University of Hyderabad
Hyderabad, India
informlayak@gmail.com, slssp@uohyd.ernet.in

Siba K. Udgata
Department of Computer & Information Sciences
University of Hyderabad,
Hyderabad, India
udgatacs@uohyd.ernet.in

***Abstract*—This paper presents a hybrid algorithm by integrating particle swarm optimization technique with stochastic ranking for solving standard constrained benchmark functions defined in CEC 2006. The problem independent characteristics of stochastic ranking and faster convergence of particle swarm optimization technique are used in the proposed hybrid stochastic ranking particle swarm optimization (SRPSO) algorithm. Performance comparison of SRPSO algorithm with five other state-of-the art techniques shows the effectiveness of the proposed hybrid algorithm. Performance comparison is based on quality of solution and robustness.**

I. Introduction

Most of the real world science and engineering optimization problems are complex and nonlinear. They have number of linear and/or nonlinear constraints embedded in it. The conventional optimization methods tend to trap in local minima while solving complex non-linear functions and thus could not solve such problems with desired accuracy level. During the last few decades, Swarm Intelligence (SI) and Evolutionary Algorithms (EA) have emerged as an alternative for such complex science and engineering design problems.

Constraint-handling techniques are mainly based on three different methods i.e.,repair method, tournament selection method and penalty method [1], [2]. The major challenge for EA and SI in Constrained Optimization Problems (COP) is to handle constraints efficiently. The penalty function is a most common approach for handling constraints. Though it is simple, the main issue is the selection of penalty factor. The optimum value of penalty factor maintains the proper balance between objective function and penalty function. As penalty factor is mostly problem-dependent [3], it is difficult to decide the optimum value apriori. Stochastic ranking technique has been proposed [4] to maintain the required balance between objective function and penalty function. This technique uses stochastic bubble-sort algorithm to rank the individuals for generating offsprings for the next generation.

In the literature, many techniques have been proposed to solve CEC 2006 benchmark functions [5], [6], [7], [8], [9]. Amirjanov et al., proposed a changing range genetic algorithm (CRGA) in which the size of the search space of feasible region adaptively shifts and shrinks by employing feasible and infeasible solutions in the population to reach the global optimum. [5]. This algorithm has many control parameters and efficiency of this method mainly depends on proper selection of these parameters. Tessema et al. proposed a SAPF technique using genetic algorithm for solving constrained optimization problems. It uses self adaptive penalty function which encourages infeasible individuals with low objective function value and low constraint violation [6]. Huang et al., proposed coevolutionary differential evolution (CDE) technique to solve the constrained problems. In this technique, a special penalty function is designed to handle the constraints and then a coevolution model is presented and differential evolution (DE) is employed to perform evolutionary search in spaces of both solutions and penalty factors [7]. Krohling et al., proposed coevolutionary particle swarm optimization with Gaussian distribution (CPSO-GD) to solve constrained optimization problems formulated as min-max problems. Montes et al., presents a simple multimembered evolution strategy (SMES) to solve global nonlinear optimization problems. Instead of using penalty functions, it uses a simple diversity mechanism to allow infeasible solutions to remain in the population [9].

Particle Swarm Optimization (PSO) is a population-based SI technique that simulates the social behavior of a group of simple individuals i.e. bird flock or fish school [10]. Based on the collective intelligence, the swarm adjusts its flying direction and search for a global optimum. PSO has shown a good performance in solving nonlinear unconstrained optimization problems [11], [12]. However the basic PSO, like other evolutionary algorithms, lacks an explicit mechanism to handle constraints which are often found in science and engineering optimization problems. In order to solve complex constrained problems, this paper proposes a hybrid algorithm by integrating stochastic ranking with PSO [4] technique, and hence named as stochastic ranking PSO (SRPSO).

This paper is organized as follows. Section II defines constrained optimization problem. Section III presents the stochastic ranking. Section IV describes the proposed stochastic ranking particle swarm optimization (SRPSO) technique. Section V reports its performance for 15 benchmark functions along with results and discussions followed by conclusions in Section 6.

II. Constrained optimization

Constrained Optimization Problems (COP) are the problems to minimize or maximize the object function under certain given constraints such as inequality, equality, upper bound and lower bound. This can be formulated as

$$\begin{aligned} Minimize\ \ f(\overrightarrow{x}), \overrightarrow{x} = (x_1, x_2, \ldots, x_D) \in S \\ subject\ \ to: \qquad g_i(\overrightarrow{x}) \leq 0, i = 1, 2, \ldots, q \\ h_i(\overrightarrow{x}) = 0, i = q+1, q+2, \ldots, m \\ l_i \leq x_i \leq u_i, i = 1, 2, \ldots, D \end{aligned} \tag{1}$$

Where f is the objective function, g_i, and h_i are the number of inequality and equality constraints respectively. The values l_i and u_i for $1 \leq i \leq D$ are the lower and upper bounds defining the search space S.

III. Stochastic Ranking Particle Swarm Optimization (SRPSO) for constrained optimization

A. Stochastic Ranking

The most widely used constrained handling method is the penalty function method. The constrained optimization problem of equation (1) can be transformed into unconstrained optimization problem with the introduction of penalty factor as

$$\psi(x) = f(x) + r_k \phi(g_i(x); i = 1, 2, \ldots, m) \tag{2}$$

where $\phi \geq 0$ is a real valued function that imposes a penalty. The penalty on each constraint is imposed by the penalty factor r_k. Although the above penalty method works well for certain constrained optimization problems, but selecting the penalty factor r_k remains to be a challenge. If the penalty factor is chosen to be too small, an infeasible solution may not be penalized enough (underpenalization), resulting a final infeasible solution. If the penalty factor is too large, a feasible solution is very likely to be found (overpenalization), but could be of very poor quality. Thus underpenalization and overpenalization are not good for in handling constraints. To solve the problem of underpenalization and overpenalization the stochastic ranking technique has been proposed [4].

The stochastic ranking technique uses a simple bubble-sort algorithm to rank the individuals for producing offsprings in the next generation. In stochastic ranking, a probability P_f is introduced to rank individuals. P_f is used to compare the objective function in infeasible regions of the search space. Generally two adjacent individuals are used for comparison. If both are in feasible space, the individual with smaller objective values will be of higher rank. If both adjacent individuals are in infeasible space, P_f is used to compare the two individuals. The individual with smaller objective value will occupy the rank by the probability P_f. If one particle is in feasible space and the other one is in infeasible space, then the particle in feasible space awarded with higher rank. Similarly all individuals are ranked by comparing adjacent individuals.

B. Particle Swarm Optimization

PSO is one of the popular SI technique being used for optimization [10]. It utilizes the searching capability of the swarm that arises from the interaction of the simple individuals [10]. Each individual (particle) in the swarm represents a potential solution. Every particle remembers its current position and the best position found so far called personal best (*pbest*). The best solution among the whole swarm is called global best (*gbest*). The location of this particle is communicated to all particles and hence the flying trajectory of the particles is recalculated based on the swarm's *gbest* t and its own *pbest* value as

$$\begin{aligned} V &= V + c_1 * \varepsilon_1 * (pbest - X) + c_2 * \varepsilon_2 * (gbest - X) \\ X &= X + V \end{aligned}$$

where X and V are position and velocity, c_1 and c_2 are cognitive and social component respectively. ε_i are independent random variables uniformly distributed in the range $[0, 1]$.

C. Stochastic Ranking Particle Swarm Optimizationtic Ranking: SRPSO

Many studies have been done for solving constrained optimization problems using EA [2], [3], [4], [13], [14], [15] and PSO [16], [17]. However, still remains a challenge to devise more efficient and effective techniques for handling constraints. PSO shows better performance on unconstrained optimization problems and popular for its fast convergence [11], [12]. However, PSO neither explicitly nor implicitly has the mechanism to handle constraints. Due to its faster convergence and effectiveness of stochastic ranking for constrained handling. SRPSO is proposed in this paper as an integration of both stochastic ranking and PSO for solving constrained optimization problems. The detail implementation steps of proposed SRPSO algorithm is explained in Algorithm 1. In the algorithm $rand_1$ and $rand_2$ are random numbers with gaussian distribution in the range [0,1]. The population of size NP are initialized randomly in the search range. The objective function $f(x)$ and penalty function $\phi(g(x))$ of all NP particles are evaluated. Based on constraint violations, particles are categorized as feasible and infeasible, and are ranked using a simple bubble-sort algorithm. The balance between underpenalization and overpenalization is achieved by setting the probability P_f less than $\frac{1}{2}$ [4]. The P_f is used to compare particles in infeasible regions of the search space. Every two adjacent particles are used for comparison, if both are in feasible space, the individual with smaller objective values will be considered as higher rank. If both the adjacent particles are in infeasible space, particle having smaller objective value assumes higher rank with probability P_f. If one particle is in feasible space and the other one is in infeasible space, then the particle in feasible space awarded with higher rank. This process is repeated until all particles are ranked. The highest ranked particle will be global best $gbest$ for the current iteration and is compared with previous $gbest$,

Algorithm 1 SRPSO

Initialization

Initialize the swarm of size NP:

Initialize position $'X'$ and velocity $'V'$ in D -dimensional search range (X_{max}, X_{min}).

Initialize $c_1 = 1.479, c_2 = 1.479$.

Evaluate the fitness of all particles (NP).

Set the current position as $pbest$ of each particle and best pbest value as $gbest$.

Optimize

for $t \leftarrow 1, Maxgen$ **do**

Update velocity and position of each particle as

$$\begin{aligned} V_{i,d}^{t+1} &= V_{i,d}^{t} + c_1 * rand_1 * (pbest_{i,d}^{t} - X_{i,d}^{t}) \\ &+ c_2 * rand_2 * (gbest_{d}^{t} - X_{i,d}^{t}) \quad (3) \\ X_{i,d}^{t+1} &= X_{i,d}^{t} + V_{i,d}^{t+1} \quad (4) \end{aligned}$$

Rank the individuals according to Stochastic Ranking.

Select the highest ranked μ particles.

Generate NP particles from μ individuals.

Evaluate fitness of each particle.

Update $pbest$: If current fitness dominates the previous then set current position as $pbest$ else retain previous $pbest$.

Update $gbest$: If best of all current $pbest$ dominates the previous $gbest$ then set current best $pbest$ as $gbest$ else retain previous $gbest$.

end for t

continue optimizing until stopping criteria or exceeding maximum iteration

Report results

Terminate

and the one with minimum objective is considered as $gbest$ for next generation. The personal best $pbest$ of the particles are also decided based on the acquired rank. The highest ranked μ individuals out of NP are selected for the next generation (μ is set as $\frac{NP}{7}$). In the next generation rest of the particles are regenerated (cloned) from highest-ranked μ particles. The learning rate and mean step sizes are set as in [4] and in our experiment P_f is set to 0.45.

IV. Simulation

The simulations are carried out using a standard PC with specifications of Pentium Core2Duo, 2GHz with 2GB RAM. Algorithm is coded in Matlab 7.2 in Windows-XP platform. The SRPSO with population size of 100 is executed for 30 independent runs on each function. We have set c_1, c_2 to be 1.479 and $P_f = 0.45$. The performance of proposed SRPSO algorithm is evaluated on 15 constrained benchmark functions chosen from CEC 2006 [18]. The chosen test problems are characterized by different difficulty levels in linear, nonlinear, quadratic, different-dimensionality and with different constraints.

V. Experimental Results and Discussion

This section presents results and discussions obtained by SRPSO on fifteen benchmark functions. The discussion is extended with the comparison of SRPSO to the current state-of-the-art techniques [5], [6], [7], [8], [9]. The complete statistical results obtained by proposed SRPSO on 15 standard benchmark function are tabulated in Table I. The first two columns of Table I show the constrained functions and their corresponding optima. This table summarizes the best, worst, median and mean results over 30 independent runs. The robustness of the algorithm is tested with standard deviation which is also tabulated in Table I along with Feasible Rate (FR). The FR denotes the percentage of the solutions which are in the feasible space. The number of violated constraints at the median solution is represented as c in Table I where the sequence of three numbers indicate the number of violations (including inequality and equalities) by more than 1.0, more than 0.01 and more than 0.0001 respectively. The mean value of violations of all constraints at the median solution is represented as v in Table I. It can be concluded from Table I that the proposed SRPSO algorithm consistently found global optima for all benchmark functions except g07 and g10 functions. SRPSO also seems to be a robust technique for constrained optimization problems as the standard deviation of solutions obtained in multiple runs is very low except g05, g06 and g10. SRPSO is compared against five recent approaches; CRGA[5], SAPF[6], CDE[7], CPSO-GD[8] and SMES[9].

Best result comparison: Table II shows the best result obtained by SRPSO for all the considered benchmark functions along with other technique. SRPSO gives comparatively good results on almost all the benchmark functions except g03.

Worst result comparison: The worst results obtained by SRPSO and other techniques are tabulated in Table III. SRPSO shows comparatively good results for g01, g05, g07, g08, g12, g15, g16, g24 benchmark functions on worst results.

Mean result comparison: Table IV shows the mean result obtained by SRPSO along with other techniques. SRPSO shows comparatively good results on most of the benchmark functions except g03, g04, g06, g09, and g10.

As shown in Tables II-IV, the proposed SRPSO method gives better result as compared to CRGA, SAPF, CDE, CPSO-GD, SMES in terms of the quality of solution and robustness as a measure of best and standard deviation values respectively.

VI. Conclusions

In this paper, we have proposed a new Stochastic Ranking based Particle Swarm Optimization algorithm (SRPSO)for constrained optimization problems. The stochastic ranking is used to strike a balance between underpenalization and overpenalization for penalty factor. The performance of proposed algorithm is evaluated for fifteen different constrained benchmark functions chosen from CEC 2006. The comparative result concludes that SRPSO is an efficient approach to solve

TABLE I
OPTIMUM RESULTS, STANDARD DEVIATION, NUMBER AND MEAN VALUE OF THE CONSTRAINED VIOLATIONS AT THE MEDIAN SOLUTION OBTAINED BY SRPSO

Func	optimum	best	median	worst	c	v	mean	std	FR
g01	-15	-15.00	-15.00	-15.00	0,0,0	0	-15.00	5.27e-12	100
g02	-0.8036191	-0.8034681	-0.7933138	-0.7572932	0,1,1	0.16	-0.7886154	1.31e-3	100
g03	-1.0005001	-0.9996365	-0.9986063	-0.9965328	0,0,0	0	-0.9984904	8.18e-5	100
g04	-30665.539	-30665.539	-30665.539	-30665.364	0,0,0	0	-30665.526	4.05e-3	100
g05	5126.4967	5126.4985	5127.6362	5145.9263	0,0,0	0	5129.9011	5.11	100
g06	-6961.8139	-6961.81397	-6961.8139	-6323.3140	0,0,0	0	-6916.1371	138.331	100
g07	24.306209	24.312803	24.360653	24.885038	0,1,1	0	24.38	1.13e-2	100
g08	-0.0958250	-0.0958250	-0.0958250	-0.0958250	0,0,0	0	-0.0958250	2.80e-11	100
g09	680.63006	680.63043	680.65104	680.76645	0,0,0	0	680.66052	3.33e-3	100
g10	7049.248	7076.397	7262.878	8075.923	1,2,2	522	7340.69640	255.37	100
g11	0.75	0.75	0.75	0.75	0,1,1	0	0.75	9.44e-5	100
g12	-1	-1	-1	-1	0,0,0	0	-1	2.62e-11	100
g15	961.71502	961.71517	961.71710	961.77126	0,0,0	0	961.72076	1.12e-2	100
g16	-1.9051553	-1.9051553	-1.9051553	-1.9051553	0,0,0	0	-1.9051553	1.12e-11	100
g24	-5.5080133	-5.5080133	-5.5080133	-5.5080133	0,0,0	0	-5.5080133	2.69e-11	100

TABLE II
COMPARISON OF BEST RESULTS OF SRPSO WITH OTHER STATE-OF-THE-ART TECHNIQUES

Func	SRPSO	CRGA[5]	SAPF[6]	CDE[7]	CPSO-GD[8]	SMES[9]
g01	-15.00	-14.9977	-15.00	-15.00	-15.00	-15.00
g02	-0.80346805	-0.802959	-0.803202	-0.794669	NA	-0.803601
g03	-0.9997	-0.9997	-1.000	NA	NA	-1.000
g04	-30665.539	-30665.520	-30665.401	-30665.539	-30665.539	-30665.539
g05	5126.4985	NA	NA	NA	NA	NA
g06	-6961.8139	-6956.251	-6961.046	-6961.814	NA	-6961.814
g07	24.312803	24.882	24.838	NA	24.711	24.327
g08	-0.095825041	-0.095825	-0.095825	NA	NA	-0.095825
g09	680.63004	680.726	680.773	680.771	680.678	680.632
g10	7076.397	7114.743	7069.981	NA	7055.6	7051.903
g11	0.75	0.750	0.749	NA	NA	0.75
g12	-1	-1.000000	-1.000000	-1.000000	NA	-1.000
g15	961.7151	NA	NA	NA	NA	NA
g16	-1.9051553	NA	NA	NA	NA	NA
g24	-5.5080133	NA	NA	NA	NA	NA

TABLE III
COMPARISON OF WORST RESULTS OF SRPSO WITH OTHER STATE-OF-THE-ART

Func	SRPSO	CRGA[5]	SAPF[6]	CDE[7]	CPSO-GD[8]	SMES[9]
g01	-15.000000	-14.9467	-13.097	-15.0000	-14.994	-15.000
g02	-0.757293	-0.722109	-0.745712	-0.779837	NA	-0.751322
g03	-0.9965328	-0.9931	-0.887	NA	NA	-1.000
g04	-30665.3644038	-30660.313	-30656.471	-30665.509	NA	-30665.539
g05	5145.9262874086544	NA	NA	NA	NA	NA
g06	-6323.3140341	-6077.123	-6943.304	-6901.285	NA	-6952.482
g07	24.885038	27.381	33.095	NA	27.166	24.843
g08	-0.095825041	-0.095808	-0.092697	NA	NA	-0.095825
g09	680.76645813	682.965	682.081	685.144	681.371	680.719
g10	8075.923755	10826.09	7489.406	NA	11458	7638.366
g11	0.750364	0.757	0.757	NA	NA	0.75
g12	-1.000000	-1.000000	-0.999548	-1.000000	NA	-1.000
g15	961.77126304593492	NA	NA	NA	NA	NA
g16	-1.9051552584358129	NA	NA	NA	NA	
g24	-5.508013271495793	NA	NA	NA	NA	NA

TABLE IV
COMPARISON OF MEAN RESULTS OF SRPSO WITH OTHER STATE-OF-THE-ART

Func	SRPSO	CRGA[5]	SAPF[6]	CDE[7]	CPSO-GD[8]	SMES[9]
g01	-15.000000	-14.9850	-14.552	-15.0000	-14.997	-15.000
g02	-0.788615	-0.764494	-0.755798	-0.785480	NA	-0.785238
g03	-0.9985	-0.9972	-0.964	NA	NA	-1.000
g04	-30665.526026	-30664.398	-30665.922	-30665.536	NA	-30665.539
g05	5129.9010819813429	NA	NA	NA	NA	NA
g06	-6916.1370274885467	-6740.288	-6953.061	-6960.603	NA	-6961.284
g07	24.38	25.746	27.328	NA	25.709	24.475
g08	-0.095825041366072852	-0.095819	-0.095635	NA	NA	-0.095825
g09	680.66052250760072	681.347	681.246	681.503	680.7810	680.643
g10	7340.6964029884484	8785.149	7238.964	NA	8464.2	7253.047
g11	0.75	0.752	0.751	NA	NA	0.75
g12	-1.000000	-1.000000	-0.99994	-1.000000	NA	-1.000
g15	961.72076523564249	NA	NA	NA	NA	NA
g16	-1.9051552584503268	NA	NA	NA	NA	NA
g24	-5.508013271537056	NA	NA	NA	NA	NA

constrained optimization problems in comparison to other popular reported techniques. This technique can be used for solving costrained engineering design problems. In future we propose to compare the computational complexity of SRPSO with other techniques to evaluate the convergence rate of the algorithm.

REFERENCES

[1] Z. Michalewicz and M. Schoenauer, "Evolutionary algorithms for constrained parameter optimization problems," *Journal of Evolutionary Computation*, vol. 4, no. 1, pp. 1–32, 1996.

[2] C. Coello and A. Carlos, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art," *Comput. Meth. Appl. Mech. Eng.*, vol. 191, pp. 1245–1287, 2002.

[3] A.E.Smith and D. Coit, *Constraint Handling Techniques Penalty Functions*. Oxford University Press and Institute of Physics Publishing, 1998.

[4] T.P.Runarsson and X.Yao, "Stochastic ranking for constrained evolutionary optimization." *IEEE Trans. Evol. Comput.*, vol. 4, no. 3, pp. 284–294, 2000.

[5] A. Amirjanov, "The development of a changing range genetic algorithm," *Computer Methods in Applied Mechanics and Engineering*, vol. 195, pp. 2495–2508, 2006.

[6] B. Tessema and G. Yen, "A self adaptive penalty function based algorithm for constrained optimization." in *IEEE Congress on Evolutionary Computation, 2006*, Vancouver, BC, Canada, July 2006, pp. 246–253.

[7] Z. Huang, F., L. Wang, and Q. He, "An effective co-evolutionary differential evolution for constrained optimization," *Applied Mathematics and Computation*, vol. 186, no. 1, pp. 340–356, 2007.

[8] A. Krohling, R. and S. Coelho, "Coevolutionary particle swarm optimization using gaussian distribution for solving constrained optimization problems," *IEEE Transactions on Systems, Man, Cybernetics*, vol. 36, no. 6, pp. 1407–1416, 2006.

[9] E. Mezura-Montes and C. A. C. Coello, "A simple multimembered evolution strategy to solve constrained optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 1, pp. 1–7, 2005.

[10] J. Robinson and Y. R. Samii, "Particle swarm optimization in electromagnetic," *IEEE Transactions on Antenna and Propagation*, vol. 52, no. 2, pp. 397–400, 2004.

[11] S. L. Sabat, L. Ali, and S. K. Udgata, "Adaptive accelerated exploration particle swarm optimizer for global multimodal functions," in *World Congress on Nature and Biologically Inspired Computing*, Coimbatore, India, December 2009, pp. 654 – 659.

[12] S. L. Sabat and L. Ali, "The hyperspherical acceleration effect particle swarm optimizer," *Appl. Soft Computing*, vol. 9, no. 13, pp. 906–917, 2008.

[13] Z. Michalewicz, "A survey of constraint handling techniques in evolutionary computation methods," in *4th Annual Conference on Evolutionary Programming*, MIT Press, San Diego, pp. 135–155.

[14] C. Coello and A. Carlos, "Use of a self-adaptive penalty approach for engineering optimization problems," *Comput. Ind.*, vol. 41, no. 2, pp. 113–127, 2000.

[15] S. Koziel and Z. Michalewicz, "Evolutionary algorithms, homomorphous mappings, and constrained parameter optimization," *Evol. Comput.*, vol. 7, no. 1, pp. 19–44, 1999.

[16] B. Yang, Y. Chen, Z. Zhao, and Q. Han, "A master-slave particle swarm optimization algorithm for solving constrained optimization problems," in *6th Congress on Intelligent Control and Automation*, Dalian, pp. 3208–3212.

[17] T.Takahama and S. Sakai, "Solving constrained optimization problems by the ϵ constrained particle swarm optimizer with adaptive velocity limit control." in *IEEE Congress on Evolution Computation*, Vancouver, BC, Canada, July 2006, pp. 308–315.

[18] J.Liang, J. Runarsson, T. P. Mezura-Montes, M. Clerc, P. Suganthan, C. Coello, and K. Deb, "Problem definitions and evaluation criteria for the CEC 2006, special session on constrained real-parameter optimization," Nanyang Technological University, Singapore, Tech. Rep., 2006.