2016 International Conference on Information System and Artificial Intelligence

An Adaptive Stochastic Ranking Mechanism in MOEA/D for Constrained Multi-objective Optimization

Wei-qin YING, Wei-peng HE, Yan-xia HUANG, Da-tong LI
School of Software Engineering
South China University of Technology
Guangzhou 510006, China
Email: yingweiqin@scut.edu.cn

Yu WU
School of Computer Science and Educational Software
Guangzhou University
Guangzhou 510006, China
Email: wuyu@gzhu.edu.cn

***Abstract*—Stochastic ranking is a promising technique for handling constraints in constrained evolutionary optimization. In stochastic ranking, the comparison between a pair of individuals is randomly based on either their objective values or degrees of constraint violation according to a constant probability parameter. In this paper, an adaptive stochastic ranking mechanism is presented to solve constrained multi-objective optimization problems (CMOPs) more effectively in the framework of the multi-objective evolutionary algorithm based on decomposition (MOEA/D). This mechanism dynamically controls the probability parameter according to both the current evolutionary stage and the difference between the degrees of constraint violation of individuals. Additionally, an elitist archiving scheme is introduced to preserve the so-far best feasible solution for each scalar sub-problem in the MOEA/D. Experimental results on CTP-series test instances show better and competitive performance of the proposed algorithm in terms of both convergence and diversity of solutions.**

Keywords-multi-objective optimization; evolutionary algorithm; constraint handling; stochastic ranking; decomposition

I. INTRODUCTION

Constrained multi-objective optimization problems (CMOPs) are frequently encountered in the real-world applications, especially in the fields of scientific research and engineering [1], [2]. Besides of multiple conflicting objectives, the difficulty in solving these problems lies in the fact that the limitations of various constraints need to be taken into account. For instance, when deploying a wireless sensor network (WSN), the task of selecting the optimal geographical positions of its sensor nodes, referred to as a WSN layout problem, can be treated as a CMOP. It aims at obtaining a network with the minimum number of sensors and the maximum lifetime under the constraint of full monitoring coverage.

Evolutionary algorithms (EAs) are suitable and have been widely used to solve multi-objective optimization problems because they have ability to find an approximation of the entire set of non-dominated solutions in the form of a population in a single run, instead of a series of separate runs. One of the most popular multi-objective EAs (MOEAs) is the non-dominated sorting genetic algorithm (NSGA-II) [3], which is based on the concept of Pareto dominance. In recent years, several decomposition-based MOEAs, such as the multi-objective evolutionary algorithm based on decomposition (MOEA/D) [4] and the conical area evolutionary algorithm (CAEA) [5], exhibit more excellent performance than dominance-based MOEAs. The MOEA/D decomposes a multi-objective optimization problem into a number of different single objective optimization sub-problems and then all these sub-problems are optimized simultaneously with the help of their respective neighbor sub-problems.

When solving CMOPs by the MOEA/D, an extra constraint handling technology needs to be adopted in its framework. The penalty function method, which uses penalty coefficients to balance objective and penalty functions, is frequently employed for handling constraints in constrained optimization problems. However, it is often quite hard to find appropriate penalty coefficients to achieve a good balance since they are problem-dependent. Stochastic ranking (SR) [6], [7], [8] is a promising constraint handling technique that avoids penalty coefficients. In this approach, a probability parameter p_f is used to determine to compare either objective values or degrees of constraint violation of a pair of given individuals for balancing objective and penalty functions directly and explicitly. In other words, if a pair of given individuals are both feasible, the objective function values are compared and the one with better objective value is regarded as better. Otherwise, when both of the given individuals are infeasible or one individual is feasible but the other is infeasible, the probability parameter p_f decides to adopt either objective values or degrees of constraint violation to measure which individual is better. More specifically, if a real number r randomly generated between 0 and 1 is less than p_f, then the one with better objective value is regarded as better; otherwise, the one with a smaller degree of constraint violation does.

However, it seems a bit unreasonable to employ a constant probability value p_f at different stages of evolution, which may lead to the fact that the algorithm converges slowly or even converges to the infeasible

978-1-5090-1585-6/16 $31.00 © 2016 IEEE
DOI 10.1109/ISAI.2016.10

regions of the search space. As a consequence, an adaptive stochastic ranking mechanism is proposed in this paper to solve CMOPs more effectively by introducing an adaptive stochastic ranking policy as well as an elitist archiving scheme.

II. ADAPTIVE STOCHASTIC RANKING MECHANISM

In this section, an adaptive stochastic ranking mechanism is designed for solving CMOPs in the MOEA/D. An adaptive stochastic ranking policy is at first introduced to improve searching ability while an elitist archiving scheme is adopted to preserve one elitist individual for each scalar sub-problem during evolution.

A. Adaptive Stochastic Ranking Policy

The algorithm might converge to infeasible regions of the search space as a result of complete lack of concern over constraints. Conversely, over-emphasis on constraints might lead to poor quality of solutions due to the loss of certain helpful information retained in infeasible individuals. Consequently, an adaptive stochastic ranking policy is proposed to handle constraints reasonably in different stages of evolution by adaptively adjusting the probability parameter p_f. By borrowing the idea of metropolis acceptance criterion, the most common acceptance probability function used in simulated annealing algorithm, the adaptive stochastic ranking function is designed in the following form:

$$P_f(\Delta E(x, y), T) = 1 - 1/(1 + e^{-|\Delta E(x,y)|/T}). \quad (1)$$

In equation (1), $\Delta E(x, y) = E(y) - E(x)$, $E(x)$ is the energy of individual x, and T is the current temperature. According to the rule of stochastic ranking, the probability parameter p_f is adopted only when at least one of two given individuals is infeasible. Thus, the difference between degrees of constraint violation can be considered as the main factor affecting the probability parameter. Therefore, the energy function is designed as the degree of constraint violation of each individual:

$$E(x) = CV(x). \quad (2)$$

Then the temperature cooling schedule is implemented in the following form:

$$T(t) = T_0/(1 + t/L). \quad (3)$$

In equation (3), t is the current generation of evolution, L is a constant sensibility coefficient of temperature, and T_0 is an initial temperature value.

As a result, two important factors influencing the value of p_f are the degrees of constraint violation and the current generation of evolution. On one hand, as for the degrees of constraint violation, the feasible individuals have the lowest energy while the infeasible individuals with high constraint violation have high energy. When the degrees of constraint violation of the given individuals are close to each other, the adaptive stochastic ranking policy applies a probability p_f close to 0.5 to compare the objective values so that the one with the worse constraint violations has a chance to survive. When the degrees of constraint violation differ greatly, the adaptive stochastic ranking policy applies a high probability, $1 - p_f$, to compare the degrees of constraint violation so that most of individuals with very high constraint violations are eliminated. On the other hand, as for the generation of evolution, the temperature T decreases with the increase of the generation of evolution and p_f tends to get a smaller value at a lower temperature. That is, at the early stage of evolution, the relatively large values of p_f give the algorithm the opportunities to explore infeasible regions of the search space and make the most use of helpful information retained in infeasible individuals. As the evolution proceeds, the probability parameter decreases according to the adaptive probability function so that the search mainly focuses on feasible regions at the terminal stage of evolution and the population finally approximates the global feasible Pareto front.

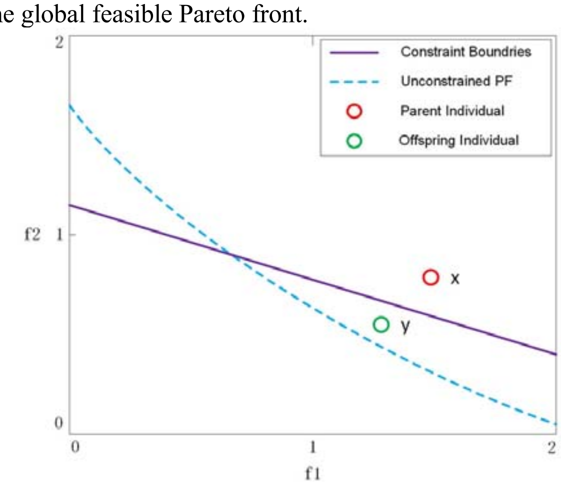

Figure 1.An illustration of elitist archiving scheme

B. Elitist Archiving Scheme

The adaptive stochastic ranking policy enables infeasible individuals to have an opportunity to replace feasible individuals during the evolution, especially at the early stage of evolution. This leads to the situation that a good feasible solution obtained previously for a scalar sub-problem might be eliminated later. For example, in the case as shown in Fig. 1, an infeasible offspring individual y with an objective vector (1.3, 0.5) is trying to update a feasible parent individual x with an objective vector (1.6, 0.7) for one sub-problem. Since y is infeasible, y cannot directly replace x and then the value p_f calculated by the adaptive stochastic ranking policy need to be compared with a random number r between 0 and 1. If $r<p_f$, the scalar aggregation objective values of two individuals for this sub-problem are compared. It comes out that y replaces x due to the better objective values although y is infeasible. In this case, a very good feasible individual x

was obtained once but is lost finally. To avoid this situation, an elitist archiving scheme is adopted in this paper. The elitist archiving scheme introduces an additional population, referred to as elitist archive, to preserve the feasible individual with the best objective value found so far for each sub-problem, referred to as elitist individual. The elitist archive doesn't participate in evolution and is updated only when a feasible offspring individual has a better objective value than the elitist individual for the corresponding sub-problem in the archive. At the end of evolution, the elitist archive is considered as the output population.

Algorithm 1 Elitist Archiving Scheme

```
 1: function UPDATEELITISTARCHIVE(y, j)
 2:     if CV(y) = 0 then
 3:         if CV(Ax^j) = 0 then
 4:             if g^te(y| j, z) ≤ g^te(Ax^j | j, z) then
 5:                 Ax^j ← y
 6:             end if
 7:         else
 8:             Ax^j ← y
 9:         end if
10:     end if
11: end function
```

The pseudo-code of the elitist archiving scheme is given in Algorithm 1. According to the elitist archiving scheme, the elitist individual Ax^j is replaced with the offspring individual y in Line 5 when both individuals are feasible and the objective value of y is better than that of Ax^j, or in Line 8 when the offspring individual y is feasible while the elitist individual Ax^j is not.

Algorithm 2 The main procedure of MOEA/D-ASR

```
 1: λ← INITIALIZEWEIGHTVECTOR()
 2: B ← INITIALIZENEIGHBORHOOD(λ)
 3: P ← INITIALIZEPOPULATION(N)
 4: z ← INITIALIZEUTOPIANPOINT(P )
 5: gen ← 0
 6: while gen ≤ MaxGen do
 7:     for i=1 to N do
 8:         parents ← SELECTIONDE(P, i)
 9:         child ← CROSSOVERDE(parents)
10:         child ← MUTATIONDE(parents)
11:         EVALUATEFITNESS(child)
12:         UPDATEUTOPIANPOINT(child)
13:         UPDATENEIGHBOR(parents, child, gen, λ , z)
14:     end for
15: end while
```

III. PROCEDURE OF ADAPTIVE SR MECHANISM

The procedure of MOEA/D using the above adaptive stochastic ranking mechanism, referred to as MOEA/D-ASR, is presented in Algorithm 2. The differential evolution (DE) operator [8], [9] is employed to generate one offspring solution in Line 9 and 10. It picks out three parent solutions from the whole population with a probability in Line 8 in order to improve the exploration ability of the search. The parent solutions are updated by the procedure UPDATENEIGHBOR in Line 13 where the maximal number of solutions replaced by a better child solution is limited to avoid trapping in local optimum.

The pseudo-code of the neighbor update rule of MOEA/D-ASR is given in Algorithm 3. The adaptive stochastic ranking mechanism is utilized in the neighbor update rule. During the update loop in Line 3-27, an offspring y tries to update at most n_r neighbor solutions. The probability parameter p_f is generated according to the adaptive stochastic ranking policy in Line 9-11. The flag variable *isUpdate* is initialized in Line 8 to trace whether the neighboring solution is updated or not. If *isUpdate* equals true after update, the elitist archiving scheme is used to update the elitist archive in Line 24, whose pseudo-code is given in Algorithm 1.

Algorithm 3 Neighbor Update Rule

```
 1: function UPDATENEIGHBOR(parents, child, gen, λ,z)
 2:     c ← 0
 3:     if c = n_r or parents = Ø then
 4:         break
 5:     else
 6:         randomly pick an solution x^j from parents
 7:         r ← rand(0,1)
 8:         isUpdate ← false
 9:         T ← T_0/(1 + gen/L)
10:         Delta ← CV (x^j ) − CV (y)
11:         P_f ← 1 – 1/(1 + exp(−|Delta|÷T ))
12:         if CV (y) = CV (x^j) = 0 or r < P_f then
13:             if g^te(y| j , z) ≤ g^te(x^j | j, z) then
14:                 x^j ← y
15:                 c = c + 1
16:                 isUpdate ← true
17:             end if
18:         else if CV (y) < CV (x^j) then
19:             x^j ← y
20:             c = c + 1
21:             isUpdate ← true
22:         end if
23:         if isUpdate = true then
24:             UPDATEELITISTARCHIVE(y,j)
25:         end if
26:         remove x^j from parents and go to step 3
27:     end if
28: end function
```

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In our experiments, MOEA/D-ASR is tested on CTP-series test instances [10] and it is compared with the MOEA/D-SR [8] as well as a variant without the elitist archiving scheme of MOEA/D-ASR, referred to as MOEA/D-ASR^{-}. The selection strategy and the operators of crossover and mutation used in each algorithm are the same as those used in [8]. The population size N = 200, the neighborhood size T=20, the maximal number of replacement n_r=2, the initial temperature T_0=1, and the temperature sensibility coefficient L=1. Besides, the probability parameter p_f used in MOEA/D-SR is set as 0.05, which is the same as that in [8]. For the sake of fairness, a total of 30 statistically independent runs of each algorithm have been conducted for each problem and each run terminates after the evolution of 200 generations. In our experiments, the set coverage metric (SC-metric) [4] and the hyper-volume metric (HV-metric) [1] are applied to assess the quality of solution set discovered by each algorithm for each benchmark problem.

TABLE I THE AVERAGE SET COVERAGE BETWEEN MOEA/D-SR (INDICATED BY SR), MOEA/D-ASR^{-} (INDICATED BY ASR^{-}) AND MOEA/D-ASR (INDICATED BY ASR) ON CTP-SERIES TEST INSTANCES.

TEST INSTANCE	C(ASR^{-}, SR)	C(SR, ASR^{-})	C(ASR, ASR^{-})	C(ASR^{-}, ASR)
CTP1	**0.071**	0.069	**0.153**	0.013
CTP2	**0.328**	0.321	**0.779**	0.020
CTP3	**0.509**	0.499	**0.940**	0.042
CTP4	**0.474**	0.354	**0.846**	0.090
CTP5	**0.470**	0.420	**0.900**	0.033
CTP6	**0.347**	0.045	**0.167**	0.040
CTP7	**0.083**	0.031	**0.069**	0.067
CTP8	**0.429**	0.030	**0.155**	0.071

Table I shows the average of the SC-metric values of the final non-dominated fronts obtained by MOEA/D-SR, MOEA/D-ASR^{-} and MOEA/D-ASR for the CTP-series test instances. It is very clear from Table I that the fronts found by MOEA/D-ASR^{-} are better than those found by MOEA/D-SR for all of the test instances, and the fronts found by MOEA/D-ASR are far better than those found by MOEA/D-ASR^{-} for all of the test instances in terms of the SC-metric. Taking CTP5 for example, 47.0% of the non-dominated solutions found by MOEA/D-SR are dominated by those of MOEA/D-ASR^{-} while 42.0% vice versa. Furthermore, 90.0% of the non-dominated solutions found by MOEA/D-ASR^{-} are dominated by those of MOEA/D-ASR while only 3.3% vice versa.

Fig. 2 presents the evolution of the average HV-metric values versus generation in MOEA/D-SR, MOEA/D-ASR^{-} and MOEA/D-ASR for CTP1-CTP8. From Fig. 2, it is very clear that MOEA/D-ASR obtains higher HV-metric values than MOEA/D-SR and MOEA/D-ASR^{-} do on most of the test instances, especially CTP3, CTP4 and CTP5.

Fig. 3 plots, in the objective space, the distribution of the final non-dominated fronts with the best HV-metric values obtained by each algorithm for CTP4 and CTP5. In addition, the unconstraint Pareto fronts are presented as blue dashed line and the constraint boundaries are presented as green solid line in Fig. 3. It is evident that MOEA/D-ASR consistently approximates well the Pareto fronts of both test instances and it achieves more convergent and uniform final fronts than the other two algorithms.

However, it can be observed from Fig. 3(a) that the final front obtained by MOEA/D-ASR^{-} for CTP4 loses the middle portion of the true front while MOEA/D-SR finds a relatively complete front regardless of its poor convergence. It suggests that the adaptive stochastic ranking policy may lead to the loss of elitist solutions because they could be replaced by some infeasible individuals with better objective values. Meanwhile, it is also clear from Fig. 3(a) that the final front obtained by MOEA/D-ASR is better than that by MOEA/D-SR in terms of convergence and better than that by MOEA/D-ASR^{-} in terms of completeness. As a consequence, it can be concluded that the adaptive stochastic ranking policy improves the global searching ability of MOEA/D on CMOPs while the elitist archiving scheme avoids losing the so-far best feasible individuals for each sub-problem when updating neighbors.

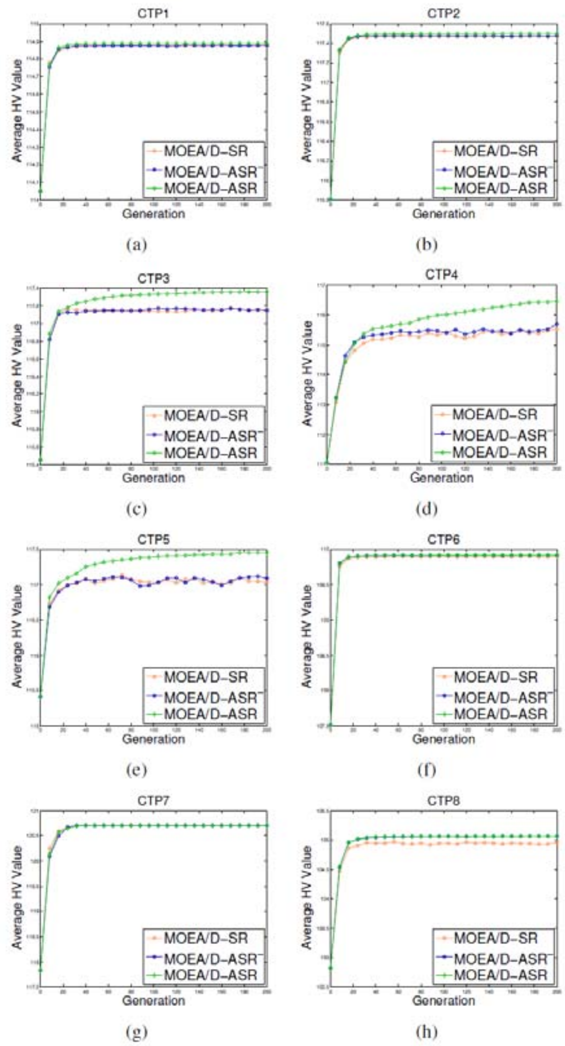

Figure 2 Evolution of the mean HV-metric values versus generation in MOEA/D-SR, MOEA/D-ASR^{-} and MOEA/D-ASR for CTP1-CTP8.

V. CONCLUSIONS

In this paper, an adaptive stochastic ranking mechanism, including an adaptive stochastic ranking policy and an elitist archiving scheme, is presented to solve constrained multi-objective optimization problems. The adaptive stochastic ranking policy can adaptively adjust the probability parameter according to both the current stage of evolution and the difference between the degrees of constraint violation of individuals by borrowing the idea of metropolis acceptance criterion. Further, the elitist archiving scheme preserves the so-far best individuals during evolution. Experimental results on CTP-series test instances verify the overall satisfactory performance of MOEA/D-ASR in terms of both convergence and diversity of solutions. Our future research will focus on applying the adaptive stochastic ranking mechanism to the CAEA for CMOPs.

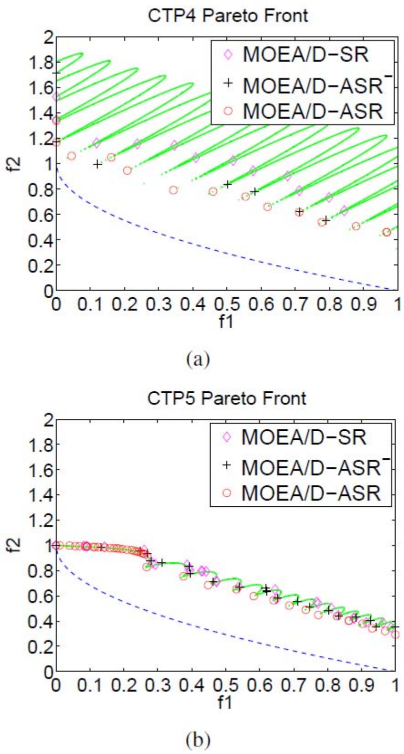

Figure 3 Plots of the nondominated fronts with the best HV-metric values in 30 runs of MOEA/D-SR, MOEA/D-ASR$^-$ and MOEA/D-ASR for CTP4 and CTP5

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (Nos. 61203310and 61503087), the Natural Science Foundation of Guangdong Province, China (No.2015A030313204), the Pearl River Science & Technology Nova Program of Guangzhou (No. 2014J2200052), the China Scholarship Council (CSC) (Nos.201406155076 and 201408440193) and the Fundamental Research Funds for the Central Universities, SCUT (No.2013ZZ0048).

REFERENCES

[1] C. A. C. Coello, D. A. Van Veldhuizen, and G. B. Lamont,*Evolutionary algorithms for solving multi-objective problems*.Springer, 2002, vol. 242.

[2] L. Jiao, J. Luo, R. Shang, and F. Liu, “A modified objectivefunction method with feasible-guiding strategy to solve constrained multi-objective optimization problems,” *Applied SoftComputing*, vol. 14, pp. 363–380, 2014.

[3] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fastand elitist multiobjective genetic algorithm: NSGA-II,” *IEEETransactions on Evolutionary Computation*, vol. 6, no. 2, pp.182–197, 2002.

[4] Q. Zhang and H. Li, “MOEA/D: A multiobjective evolutionary algorithm based on decomposition,” *IEEE Transactionson Evolutionary Computation*, vol. 11, no. 6, pp. 712–731,2007.

[5] W. Ying, X. Xu, Y. Feng, and Y. Wu, “An efficient conicalarea evolutionary algorithm for bi-objective optimization,”*IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 95, no. 8, pp. 1420–1425, 2012.

[6] T. P. Runarsson and X. Yao, “Stochastic ranking for constrained evolutionary optimization,” *IEEE Transactions onEvolutionary Computation*, vol. 4, no. 3, pp. 284–294, 2000.

[7] ——, “Search biases in constrained evolutionary optimization,” *IEEE Transactions on Systems, Man, and Cybernetics,Part C: Applications and Reviews*, vol. 35, no. 2, pp. 233–243, 2005.

[8] M. A. Jan and R. A. Khanum, “A study of two penaltyparameterless constraint handling techniques in the framework of MOEA/D,” *Applied Soft Computing*, vol. 13, no. 1,pp. 128–148, 2013.

[9] H. Li and Q. Zhang, “Multiobjective optimization problemswith complicated pareto sets, MOEA/D and NSGA-II,” *IEEETransactions on Evolutionary Computation*, vol. 13, no. 2,pp. 284–302, 2009.

[10] K. Deb, A. Pratap, and T. Meyarivan, “Constrained testproblems for multi-objective evolutionary optimization,” in*Evolutionary Multi-Criterion Optimization*. Springer, 2001,pp. 284–298.