2009 International Joint Conference on Computational Sciences and Optimization

A Multi-objective Optimization Evolutionary Algorithm Addressing Diversity Maintenance

Xiaoning Shen, Min Zhang, Tao Li
School of Information and Control
Nanjing University of Information Science and Technology
Nanjing, 210044, China
sxnystsyt@sina.com.cn

Abstract

A multi-objective evolutionary algorithm keeping diversity of the population is proposed. The algorithm makes use of a metric based on entropy to measure the diversity of the population. The evolving state of the current population is associated with the running mechanism of the algorithm by the diversity metric, and several strategies are designed to enhance the extent of exploration of the algorithm. Simulation results indicate that the proposed algorithm has good performance of convergence and distribution.

1. Introduction

Evolutionary algorithms have been recognized to be well suited for multi-objective optimization problems due to their capability to evolve a family of solutions simultaneously and efficiently in one run. Diversity maintenance is one major technical issue of the multi-objective evolutionary algorithms. So far, there exist some methods to maintain diversity within the evolving population. Fitness sharing used in MOGA [1] adjusts individuals' fitness values by the sharing function which represents similarity among individuals. In PAES [2], the density around an individual is estimated by the number of individuals in the same box of the grid. SPEA2 [3] and NSGA-□ [4] employ the k-th nearest neighbor technique and the crowding distance method respectively to estimate density in every individual's neighborhood. Algorithms using above approaches have different characteristics, while in some difficult problems, they can only generate sub-regions of the Pareto front where solutions gather together in small parts, or they may even converge prematurely onto locally optimal sets of non-dominated solutions. There are several reasons for that. 1) The density information is used as the basis for selection only when non-dominated levels of the compared individuals are equal. In this way, the algorithms only ensure diversity within the approximation set produced currently, but can not be pushed towards wider and better regions of optimality. 2) Due to the excessive exploitation of elites, individuals with better fitness values will occupy the whole population gradually. It's difficult for the breeding operators to generate solutions in the new regions because of the lack of diversity in individual patterns in the mating pool. 3) The selection and searching mechanisms are not adjusted dynamically according to the varying of the evolving state of the population, which may result in blindness of the algorithm to some extent.

To overcome drawbacks of the above-mentioned approaches, a new multi-objective evolutionary algorithm addressing diversity maintenance is proposed. Simulation results indicate that the proposed algorithm has good searching performance.

2. A diversity metric based on entropy

A diversity metric based on entropy is proposed.

Definition 1. Suppose individuals in the population pop can be divided into K different classes of subsets $C_1, C_2, \cdots, C_K$ according to the similarity among their objective vectors; the number of individuals in each subset is $|C_1|, |C_2|, \cdots, |C_K|$; $\forall i, j \in \{1,2,\cdots,K\}$, $C_i \cap C_j = \phi$ and $\bigcup_{i=1}^{K} C_i = pop$. The diversity metric E of pop is defined as:

$$E = \frac{-\sum_{j=1}^{K} P_j \log(P_j)}{D^{\max}} \tag{1}$$

978-0-7695-3605-7/09 $25.00 © 2009 IEEE
DOI 10.1109/CSO.2009.28

where $P_j = |C_j|/N$; N is the population size; P_j is the probability of an individual belonging to the *j*-th subset; $D^{\max}$ is the maximum value of the numerator, which can be obtained when $K = N$. $0 \le E \le 1$.

The method to estimate the similarity among objective vectors to classify individuals is as follows:

(1) Calculate the maximum and the minimum value $f_i^{\max}$ and $f_i^{\min}$ of *pop* on f_i, $i = 1,2,\cdots,m$.

(2) Objectives of each individual can be normalized as: $\boldsymbol{F}^{norm}(\boldsymbol{x}) = (f_1^{norm}(\boldsymbol{x}), f_2^{norm}(\boldsymbol{x}),\cdots, f_m^{norm}(\boldsymbol{x}))$, where $f_i^{norm}(\boldsymbol{x}) = (f_i(\boldsymbol{x}) - f_i^{\min})/(f_i^{\max} - f_i^{\min})$.

(3) For each pair of individuals $\boldsymbol{x}$ and $\boldsymbol{x}'$ in *pop*, if $\|\boldsymbol{F}^{norm}(\boldsymbol{x}) - \boldsymbol{F}^{norm}(\boldsymbol{x}')\| < \delta$, then they are classified into the same subset. Each subset C_j generated after classification satisfies that $\forall\, \boldsymbol{x} \in C_j$, $\exists\, \boldsymbol{x}' \in C_j$ and $\boldsymbol{x}' \ne \boldsymbol{x}$, $\|\boldsymbol{F}^{norm}(\boldsymbol{x}) - \boldsymbol{F}^{norm}(\boldsymbol{x}')\| < \delta$ holds.

The parameter δ is calculated as follows:

$$\delta = \sqrt{(1/N)^2 * m} \qquad (2)$$

The aim is to make all the N individuals distribute uniformly along each objective so that the diversity in the objective space can be maintained.

It can be seen that the more the kinds of individuals in the population, the greater the value of E .

3. A multi-objective algorithm addressing diversity maintenance

3.1. The use of an external set

The proposed algorithm ADMOEA adopts an external set to preserve the non-dominated solutions searched by the algorithm. The diversity of the current external set is measured in every generation, to change the formation of the population in the new generation.

In each generation of ADMOEA, a certain number (which is set as N_{1t}) of individuals with small density re taken out from the external set to form parts of the parent population in the new generation. Perform mutation on this part of the population and generate the corresponding child population. The aim of this operation is to make use of the good genetic characters of the elitist individuals located in the sparse region and explore more solutions with better quality or greater potential in the vicinity of them.

N_{1t} is determined by the value of the diversity metric E_t of the current external set adaptively:

$$N_{1t} = \lfloor a * (1 - E_t)^2 * N \rfloor + b \qquad (3)$$

where N is the total size of the parent population; $a = 0.5$, $b = 0.1 * N$.

When E_t is small which indicates the diversity of the current external set is bad, more individuals are selected for exploration in the less crowded areas in order to improve the distribution of the external set.

3.2. Control of the elitist individuals

A controlled elitism strategy is used to maintain the balance between the exploration of new individuals and the exploitation of elitist individuals. ADMOEA forms parts of the parent population as follows:

In the *t*-th generation, suppose the combined population is sorted into L_t non-dominated fronts, and the crowding distance is used to rank individuals of each front in descending order. A total of N_{2t} ($N_{2t} = N - N_{1t}$) individuals are selected from all the fronts to form a part of the new parent population. Assume that the maximum number of individuals allowed to be taken from the *i*-th ($i = 1,2,\cdots,L_t$) front is n_{it} and set n_{it} to be geometrical series:

$$n_{it} = r_t n_{(i-1)t}\,, i = 2,\cdots,L_t \qquad (4)$$

where $0 < r_t < 1$ is the reduction rate. It can be obtained from the geometrical formulation that:

$$n_{it} = N_{2t} * \frac{1 - r_t}{1 - r_t^{L_t}} * r_t^{i-1}\,,\ \ i = 1,2,\cdots,L_t \qquad (5)$$

This approach allows individuals from all the non-dominated fronts to co-exist in the new parent population so that the diversity of individual patterns is increased while some elitist individuals are preserved.

The value of r_t is set to change adaptively with the varying of the metric E_t of the current external set:

$$r_t = 1 - E_t^2 \qquad (6)$$

When E_t is great which indicates the diversity of the current external set is good, r_t is set to be a smaller value so that more individuals are taken from the better non-dominated fronts and fewer individuals are taken from the worse fronts to increase the exploitation of the elitist individuals.

3.3. Conversion between exploration and exploitation

Set two threshold parameters C_{good} and C_{bad} . When $E_t > C_{good}$, it's considered that the current external set has good diversity, and the algorithm enters the mode of exploitation. Better individuals are further exploited based on inheriting good genetic characters of existing individuals. When $E_t < C_{bad}$, the algorithm is converted into the mode of exploration. A certain number of random individuals are inserted into the population and the potential patterns that may exist in the random individuals are utilized to explore good individuals in the unknown regions. In this way, the searching range of the algorithm is widened and the diversity of the population is increased.

3.4. The procedure of the algorithm

Step1: Set $t=1$. Generate an initial parent population $parpop_t$ with the size of N and create an empty external set $archive_t = \phi$. The child population $chipop_t$ is generated using the breeding operators. Set *mode* = exploration, and the maximum number of generations to be T.

Step2: Set $mixedpop_t = parpop_t \cup chipop_t$. The fast non-dominated sorting procedure is applied to $mixedpop_t$. For individuals in each of the resulting non-dominated fronts, the crowding distances are calculated to rank them in descending order.

Step3: Find all the non-dominated solutions in $mixedpop_t$ and copy them into $archive_t$. Thereafter, delete all the repeated and dominated solutions in $archive_t$. If the number of solutions in $archive_t$ exceeds a given maximum M, truncate $archive_t$ by removing solutions with small crowding distances.

Step4: Compute the metric E_t of $archive_t$.

Step5: If *mode* = exploration and $E_t > C_{good}$, then *mode* = exploitation; if *mode* = exploitation and $E_t < C_{bad}$, then *mode* = exploration.

Step6: Compute N_{1t} by E_t according to the formulation (3), then compute $N_{2t} = N - N_{1t}$. If *mode* = exploration, generate a random population with the size of $0.1 * N$ and set $N_{2t} = N_{2t} - 0.1 * N$.

Step7: Select $parpop_{1(t+1)}$ with the size of N_{1t} from $archive_t$ according to Section 3.1 and perform mutation on it to generate the child population $chipop_{1(t+1)}$. Select $parpop_{2(t+1)}$ with the size of N_{2t} from $mixedpop_t$ according to Section 3.2 and perform binary tournament selection, crossover and mutation on it to generate the child population $chipop_{2(t+1)}$.

Step8: Set the new parent population $parpop_{t+1} = parpop_{1(t+1)} \cup parpop_{2(t+1)}$. If *mode* = exploration, then the child population $chipop_{t+1} = chipop_{1(t+1)} \cup chipop_{2(t+1)} \cup randpop_{t+1}$. Otherwise, $chipop_{t+1} = chipop_{1(t+1)} \cup chipop_{2(t+1)}$.

Step9: If $t \geq T$, the set of the non-dominated solutions stored in $archive_t$ is considered as the Pareto set approximation and the algorithm stops. Otherwise, set $archive_{t+1} \leftarrow archive_t$, $t = t+1$ and go to Step 2.

4. Experimental results

4.1. Description of the optimization problem

The proposed algorithm is applied to a mechanical design problem T_1 [5] to validate its efficiency.

T_1: As shown in Figure 1, the truss has to carry a certain load without elastic failure. The objectives of designing the truss are to minimize the volume of the truss and stresses in AC and BC. The optimization model of the problem is expressed as follows:

$$\begin{cases} \text{Minimize } f_1(\boldsymbol{x}) = x_1\sqrt{16+y^2} + x_2\sqrt{1+y^2}, \\ \text{Minimize } f_2(\boldsymbol{x}) = \max(\sigma_{\mathrm{AC}}, \sigma_{\mathrm{BC}}). \\ s.t.\ \max(\sigma_{\mathrm{AC}}, \sigma_{\mathrm{BC}}) \leq 10^5,\ 1 \leq y \leq 3 \text{ and } 0 \leq x_1, x_2 \leq 0.01 \end{cases} \tag{7}$$

where y is the vertical distance between B and C; x_1 and x_2 are the length of AC and BC respectively. Stresses in AC and BC are calculated as follows:

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

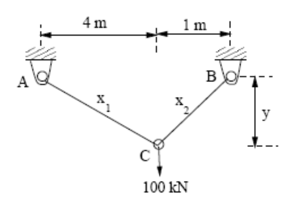

Figure 1. Illustration of the two-bar truss

4.2 Experimental results

ADMOEA is compared with two other algorithms NSGA-□ and controlled NSGA-□using the following metrics: the metric C [6], the distribution metric SP [7], the diversity metric E given in this paper, the running time t and visual graphs of Pareto fronts. The smaller the value of SP and the greater the value of E, the better the distribution and diversity of the solution set. The metric C gives a comparison of two solution sets based on their domination or equality to each other. $C(X_1, X_2) > C(X_2, X_1)$ indicates that X_1 is better than X_2 in terms of the performance of convergence.

Each of the three algorithms is performed 10 times independently in the problem T_1 , and the average results are given in Table 1 and Table 2 respectively. Comparisons of Pareto fronts are shown in Figure 2.

It can be seen that the performance of convergence (indicated by C), distribution (indicated by SP) and diversity (indicated by E) of ADMOEA is obviously better than that of the existing algorithms in T_1 . These results indicate that the strategy of guiding the selection operation and the searching mechanism adaptively based on the diversity metric adopted by ADMOEA is feasible and effective. It can widen the searching range of the algorithm and increase the diversity of the evolving population so that good individuals in the new regions can be found to jump out of the local optima and improve the convergence and distribution of the algorithm.

Table 1 indicates that the average running time of ADMOEA is longer than that of NSGA-□ and controlled NSGA-□.This is because compared to the other two algorithms, there are additional operations in ADMOEA which are update and truncation of the external set, evaluation of the diversity metric and several strategies enhancing the extent of exploration. This shows that the performance of ADMOEA is improved at the expense of time consumption.

Table 1. Comparisons on the metric SP , E , t

	SP	*E*	*t*/s
KDMOGA	0.0046	0.8789	74.3515
NSGA-□	0.0114	0.7399	58.4037
controlled NSGA-□	0.0098	0.7696	59.7748

Table 2. Comparisons on the metric C

C(A, N)	*C*(N, A)	*C*(A, c)	*C*(c, A)	*C*(N, c)	*C*(c, N)
0.3100	0.1071	0.3586	0.0729	0.2943	0.2100

Note: A denotes the proposed algorithm ADMOEA, N denotes NSGA-□, c denotes controlled NSGA-□.

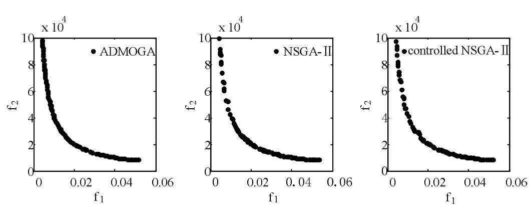

Figure 2. Comparisons of Pareto fronts in T_1

5. Conclusions

In this paper, issues related to the diversity maintenance of the population in multi-objective evolutionary algorithms are considered. A diversity metric is given based on the concept of entropy. The selection operation and the searching mechanisms of the algorithm are adjusted adaptively according to the diversity metric of the population. A new multi-objective evolutionary algorithm addressing diversity maintenance is proposed based on the above-mentioned strategies. Experimental results indicate that the proposed algorithm has better searching performance than that of the two existing algorithms.

Acknowledgements

This work was supported by a grant from National Laboratory of Space Intelligent Control.

References

[1] C.M. Fonseca., P.J. Fleming. Evolutionary algorithms for multiobjective optimization: Formulation, discussion and generalization. *In: Proc. of the Fifth International Conference on Evolutionary Algorithms*, pages 416-423, 1993.

[2] J.D. Knowles, D.W. Corne. The Pareto archived evolution strategy: A new baseline algorithm for Pareto multiobjective optimization. *In: Proc. of the Congress on Evolutionary Computation*, pages 98-105, 1999.

[3] E. Zitzler, M. Laumanns, L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm. *In: Proc. of Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*, pages 95-100, 2002.

[4] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan. A fast and elitist multiobjective evolutionary algorithm: NSGA-□. *IEEE transactions on evolutionary computation*, 6(2): 182-197, 2002.

[5] K. Deb, A. Pratap, S. Moitra. Mechanical component design for multiple objectives using elitist non-dominated sorting GA. *KanGAL report 200002, Indian Institute of Technology, Kanpur, India*, 2000

[6] E. Zitzler, K. Deb, L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, 8(2): 173-195, 2000.

[7] C.A. Coello, M. R. Sierra. A coevolutionary multi-objective evolutionary algorithm. *In: Proc. of the Congress on Evolutionary Computation*, pages 482-489, 2003.