A Fast Dynamical Evolutionary Algorithm for Multi-objective Mechanical Component Design

XIUFEN ZOU[1], MINLING WANG[1], MINZHONG LIU[2], LISHAN KANG[2]
[1]College of Mathematics and Statistics,
Wuhan University, Wuhan 430072, China,
[2]State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072,
CHINA

Abstract:-In this paper, a fast dynamical multi-objective evolutionary algorithm (DMOEA) based of the principle of the minimal free energy in thermodynamics, was developed to solve mechanical component design problems. Its performance was compared with the ε-constraint method and NSGA-II proposed by Deb et al. Simulation results show that the proposed evolutionary approach produces excellent solutions for mechanical component design problems.

Key-Words: -multi-objective optimization, evolutionary algorithm, mechanical component design.

1 Introduction

Most real-world problems are multi-objective optimization problems (MOPs), and the effectiveness of evolutionary algorithms (EAs) in solving these problems has been widely recognized in recent ten years [1]. After David Schaffer's first study on evolutionary multi-objective optimization (EMO) in the mid of 1980s [2], a number of Pareto-based techniques and elitist algorithms have been proposed in the last decade [3][4], such as Pareto-based ranking procedure (FFGA) [5], niched Pareto genetic algorithm (NPGA) [6], Pareto-archived evolution strategy (PAES) [7], nondominated sorting genetic algorithm (NSGA) [8], and NSGA II [9], the strength Pareto evolutionary algorithm (SPEA) [10], and SPEA2 [11], thermo-dynamical genetic algorithm (TDGA) [12].

Although these techniques performed well in different comparative studies, there is still a large room for improvement as recent studies have shown [9] [13] [14].

In this paper, we propose a dynamical multi-objective evolutionary algorithm (DMOEA) based on the principle of the minimal free energy in statistical physics, for solving multi-objective optimization problems (MOPs). Two new ideas are introduced in DMOEA. One is a fitness assignment strategy by combining Pareto dominance relation and Gibbs entropy. The other is density distance and the Metropolis criterion for selection of new individuals in each generation.

The paper is structured as follows: Section 2 provides rather detailed descriptions of the proposed algorithm. In Section 3, numerical experiments are conducted, two measurements proposed by Deb [9] are used to compare DMOEA with other three well-known multi-objective evolutionary algorithms (MOEAS): NSGAII, SPEA, PAES for a number of test problems, moreover, the DMOEA is applied to solve the two mechanical component design problems and comparisons with other methods are made. Finally, some conclusions and future work are addressed in Section 4.

2 Description of the DMOEA for Multi-objective Optimization

Without loss of generality, we consider the following multi-objective minimization problem with n decision variables (parameters), M objectives and k constrained conditions:
Minimize

$$y = F(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \cdots, f_M(\vec{x})) \qquad (1)$$

subject to $\quad g_i(\vec{x}) \le 0, \ i = 1,2,\cdots,k;$

2.1 The New Fitness Assignment Strategy

In statistical physics, the Gibbs distribution models a system in thermo-dynamical equilibrium at a given temperature. Further, it is also known that this distribution minimizes the free energy F defined by

$$F = \langle E \rangle - TS \qquad (2)$$

Where $< E >$ is the mean energy of the system, S is the entropy and T is the temperature. The minimization of F means minimization of $< E >$ and maximization of TS. It is called "the principle of the minimal free energy".

Such a statistical framework has been introduced into many fields. Since the minimization of the objective function (convergence towards the Pareto-optimal set) and the maximization of diversity in obtained solutions are two key goals in the multi-objective optimization, the working principle of a MOEA and the principle of finding the minimum free energy state in a thermodynamic system is analogous. In order to plunge a multi-objective optimization problem into such a statistical framework, we combine the rank value $R(i)$ calculated by Pareto-dominance relation with Gibbs entropy $S(i)$ to assign a new fitness $F(i)$ for each individual i in the population, that is

$$F(i) = R(i) - TS(i) \qquad (3)$$

where $R(i)$ is the rank value of individual i, which is equal to the number of solution n_i that dominates solution i [5]. The rank values can be computed as follows.

$$R(i) = |\Omega_i|, \qquad (4)$$

where $\Omega_i = \{\vec{x}_j \mid \vec{x}_j \prec \vec{x}_i, 1 \le j \le N, j \ne i\}$ .In this way, the $R(i)=0$ corresponds to a non-dominated individual, while a high $Rank(i)$ means that i is dominated by many individuals.

$$S(i) = -p_T(i)\log p_T(i) \qquad (5)$$

where $p_T(i) = \frac{1}{Z}\exp(-\frac{R(i)}{T})$ is the analogue of the Gibbs distribution, $Z = \sum_{i=1}^{N}\exp(-\frac{R(i)}{T})$ is called the partition function, and N is the population size.

2.2 Density Estimation Technique

From the expression (3), we easily observe that it is difficult to distinguish the different individuals when their Rank values are equal. Therefore, we use a density estimation technique (which proposed by Deb et al[9]) to compute an individual's crowding distance. The crowding distance $d(i)$ for individual i is calculated according to the following steps:

(1) Sorting the solution set I according to each objective function in ascending order of magnitude.

(2) Noted that $d_m(i)$ is the crowding distance of individual i referring to the m-th objective function, then $d_m(1) = d_m(l) = \infty$ (the boundary solutions are assigned an infinite distance value), $d_m(i) = \frac{f_m^{i+1} - f_m^{i-1}}{f_m^{\max} - f_m^{\min}}$, $i = 2,3,\cdots,l-1$, where f_m^i is the m-th objective function, value of the i-th individual in the set I, $f_m^{\max}$ and $f_m^{\min}$ are the maximum and minimum values of the m-th objective function.

(3) Calculating the sum corresponding to all objective functions, $d(i) = \sum_{m=1}^{M} d_m(i)$, where M is the number of all objective functions.

A solution with a smaller value of this distance measure is, in some sense, more crowded by other solutions. So we use the crowding distance to correct the expression of fitness value (3):

$$fitness(i) = R(i) - TS_i - d(i) \qquad (6)$$

In DMOEA, the fitness values are sorting in increasing order. The individual in population that the fitness value is smallest is called "the best individual", and the individual in population that the fitness value is largest is called "the worst individual".

2.3 The New Selection Criterion

In every generation, we always obtain new individuals by genetic operator, but it is worthwhile discussing that we use what kind of way to accept the new individuals, or eliminate the old individuals, and form new population at next generation. Since the DMOEA is based on the thermodynamical principle, we attempt to employ the Metropolis criterion of simulated annealing algorithm (SA)[15] and the crowding distance to guide the select process, that is,

(1) If $R(X_{new}) < R(X_{worse})$, then $X_{worst} := X_{new}$

(2) If $R(X_{new}) = R(X_{worse})$ and $d(X_{new}) > d(X_{worse})$, then $X_{worst} := X_{new}$

(3) else if $\exp(\frac{R_{worst} - R_{new}}{T})$ >random (0,1), then $X_{worst} := X_{new}$

Where R_{worst} and R_{new} are respectively the Rank values of the worst individuals and the new individuals.

The structure of DMOEA is described as follows:

Procedure DMOEA

Step1: t=0, generate randomly an initial population P(t)={X_1 X_2… X_N}, N is the population size;
Step2: Calculate the rank values {R_1(t),…, R_N (t)} of all individuals in P(t)according to (4);
Step3: Save the individuals whose rank values are equal to zero;
Step4: Calculate the fitness of all individuals according to equation (6), and sort them in increasing order;
Step5: Repeatedly execute step6 to step11 until the termination conditions are satisfied;
Step6: t=t+1;
Step7:Randomly select m_1 individuals to do multi-parent crossover and m_2 individuals to mutate, and to generate n new individuals;
Step8: Compare the new individuals with the worst individuals, and accept the new individuals according to Section2.3;
Step9: Calculate the rank values {R_1(t),…, R_N (t)} of all individuals in new population P(t) according to (4);
Step10: Save the individuals whose the rank values are equal to zero;
Step11:Calculate the fitness of all individuals according to equation (6), sort them in increasing order, and record the worst individuals;
Step12: Output the all results.

Remark: In step 7, multi-parents crossover originated appeared in Guo T. et al [16].Randomly select *m* individual $X_1', X_2', \ldots, X_m'$ from population P(t) to form a new individual s , $s = \sum_{i=1}^{m} a_i X_i'$,where a_i is a random number in [-0.5,1.5], and $\sum_{i=1}^{m} a_i = 1$ (Quasi-linear combination can guarantee searching more extensive range, even including the boundary).

3 The Numerical Experiments

3.1 Test Problems

To verify the efficiency and effectiveness of the DMOEA, we have conducted many numerical experiments and compare its performance with several other MOEAs: NSGA II, SPEA,PAES.

(1) Nine Difficult test problems from Deb [3][9]

SCH1 , FON , POL, KUR , ZDT1 , ZDT2 , ZDT3,ZDT4,ZDT6
The following is the two mechanical component design problems;

(2)Two-Bar Truss Design

This problem was originally studied using the ε-constraint method [17], NSGAII [3] and Azarm, S. et al [18], John Eddy et al[19]. The design of truss (see Fig.1) can be written as the following two-objective minimization problem with three variables (y is vertical distance between B and C in m, x_1 and x_2 is the cross-sectional areas of AC and BC respectively in m^2, f_{volume} is the overall volume of material used, $f_{stress,AC}$ is the stress in bar AC). The system is subject to three inequality constraints. The first two are constraints on the objective function values, the third constraint limits the stress in bar BC.

$$\begin{aligned}
&\textit{Minimize} && f_{volume} = x_1\sqrt{16+y^2} + x_2\sqrt{1+y^2} \\
&\textit{Minimize} && f_{stress,AC} = \frac{20\sqrt{16+y^2}}{yx_1} \\
&\textit{Subject to} && g_1(x_1,x_2,y) = f_{volume} \le 0.1 \\
& && g_2(x_1,x_2,y) = f_{stress,AC} \le 100 \\
& && g_3(x_1,x_2,y) = \frac{80\sqrt{1+y^2}}{yx_2} \le 100 \\
& && 0.0001 \le x_1 \le 0.1, \\
& && 0.0001 \le x_2 \le 0.1, \\
& && 1 \le y \le 3
\end{aligned}$$

(3) Gear Train Design

A compound gear train is to be designed to achieve a specific gear ratio between the driver and driven shafts (see Fig.2, is a copy of Fig.293 from Deb [3]). The objective of the gear train design is to find the number of teeth in each of the four gears so as to minimize (a) the error between the obtained gear ratio and a required gear ratio of 1/6.931[20] and (b) the maximum size of any of the four gears. Since the number of teeth must be integers, all four variables are thus strictly integers. By denoting the variable vector $X = (x_1, x_2, x_3, x_4) = (T_d, T_b, T_a, T_f)$, we write the two-objective optimization problem as follows:

$$Minimize\ \ f_1(X) = (\frac{1}{6.931} - \frac{x_1 x_2}{x_3 x_4})^2$$

$$Minimize\ \ f_2(X) = \max(x_1, x_2, x_3, x_4)$$

$$s.t.\ 12 \le x_1, x_2, x_3, x_4 \le 60,\ all\ x_i s\ are\ integers$$

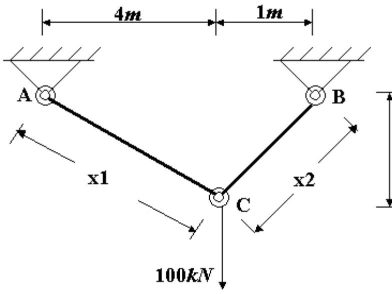

Fig.1 A two-bar truss. This is a reprint of Fig. 3 from John Eddy et al [19]

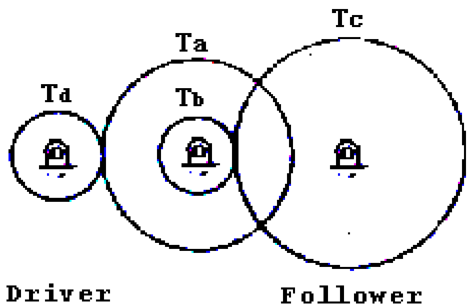

Fig.2 A compound gear train

3.2 Results and Discussion

In order to compare with NSGA II, SPEA, PAES, we use the two performance metrics defined in [9] to test DMOEA, and the setting of parameters is basically the same as that in [9](see Table 1). The algorithm has been coded in C language and implemented on a Pentium PC 500MHz in double precision arithmetic. For each problem, 20 runs with different random seeds have been carried out.

Table 2 shows the mean and variance of the convergence metric obtained by four algorithms: DMOEA, NSGA-II (real-coded), SPEA, and PAES. DMOEA is able to converge better in all problems except in POL, where NSGA-II found better convergence. In all cases with DMOEA, the variance in 20 runs is also smaller (In the table, the zero variance means that the variance is less than 10^{-6}).

Table 3 shows the mean and variance of the diversity metric obtained by all four algorithms. DMOEA performs better than NSGA-II (real-coded), SPEA and PAES in all test problems except in KUR, where NSGA-II found better diversity.

Fig.3 shows the obtained non-dominated solutions by DMOEA for the two-bar truss problem. The figure shows that DMOEA can obtain widely spread solutions when compared to the ε-constraint method and NSGAII (see ref. [3])

Table 4 shows the extreme solutions (marked as 'EE' and 'DD') obtained by DMOEA for the gear train design, which indicates the power of the DMOEA in finding a wider spread of solutions when compared to the NSGA II (the extreme solutions is marked as 'E' and 'D' [3]). What is also important is that all solutions of the NSGA II (including 'EE' and 'DD') have been found in just one simulation run of the DMOEA.

Table 1 Parameter settings

Population Size: *N=100*
Crossover: Multi-parent crossover, the crossover probability is 0.1(different from [9])
Mutate: uniform mutate, the mutate probability is *1/n* (where *n* is the number of decision variables)
The maximum generation: *g=250 (25000 function evaluations)*
Temperature: *T=10000*.

Table 2 The mean $\bar{\gamma}$ (first row) and variance σ_γ^2 (second row) of the convergence metric

Algorithm	SCH1	FON	POL	KUR	-
DMOEA	0.000408	0.002245	0.0347034	0.022262	-
	0	0	0.000043	0	-
NSGA-‖ (Real-coded)	0.003891	0.001931	0.015553	0.028964	-
	0	0	0.000001	0.000018	-
SPEA	0.003403	0.125692	0.037812	0.045617	-
	0	0.000038	0.000088	0.00005	-
PAES	0.001313	0.151263	0.030864	0.057323	-
	0.000003	0.000905	0.000431	0.011989	-
Algorithm	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
DMOEA	0.000150	0.000119	0.000493	0.000290	0.000520
	0	0	0	0.000001	0
NSGA-II (Real-codes)	0.033482	0.072391	0.114500	0.513053	0.296564
	0.004750	0.031689	0.007940	0.118460	0.013135
SPEA	0.001799	0.001339	0.047517	7.340299	0.221138
	0.000001	0	0.000047	6.572516	0.000449
PAES	0.082085	0.126276	0.023872	0.854816	0.085469
	0.008679	0.036877	0.000001	0.527238	0.006664

Table 3 The mean $\overline{\Delta}$ (first row) and variance σ_{Δ}^{2} (second row) of the diversity metric

Algorithm	SCH1	FON	POL	KUR	-
DMOEA	0.202006	0.351841	0.272671	0.580337	-
	0.000416	0.000345	0.001789	0.000297	-
NSGAII (Real-coded)	0.477899	0.378065	0.452150	0.411477	-
	0.003471	0.000639	0.002868	0.000992	-
SPEA	1.021110	0.792352	0.972783	0.852990	-
	0.004372	0.005546	0.008475	0.002619	-
PAES	1.063288	1.162528	1.020007	1.079838	-
	0.002868	0.008945	0	0.013772	-
Algorithm	ZDT1	ZDT2	ZDT3	ZDT4	ZDT6
DMOEA	0.321018	0.322464	0.558973	0.487381	0.313715
	0.001055	0.000589	0.003799	0.007740	0.000534
NSGAII (Real-coded)	0.390307	0.430776	0.738540	0.702612	0.668025
	0.001876	0.004721	0.019706	0.064648	0.009923
SPEA	0.784525	0.755148	0.672938	0.798463	0.849389
	0.004440	0.004521	0.003587	0.014616	0.002713
PAES	1.229794	1.165942	0.789920	0.870458	1.153052
	0.004839	0.007682	0.001653	0.101399	0.003916

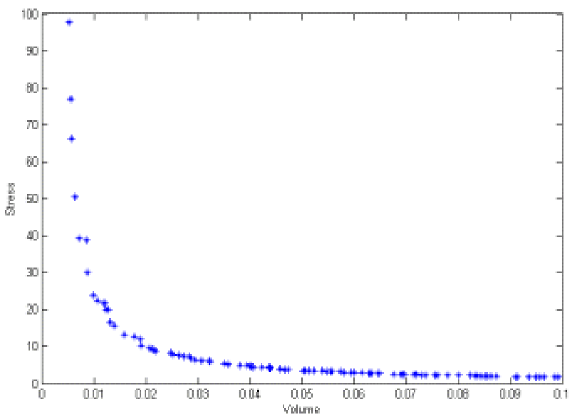

Fig.3 Optimized solutions obtained using DMOEA for the two-bar truss problem

Table 4 Obtained extreme solutions by DMOEA for the gear train design

Solution	x1	x2	x3	x4	Max.diameter(cm)	Error
EE	15	15	39	40	40	2 .36(10^{-9})
E	12	12	27	37	37	1 .83(10^{-8})
D	12	12	13	13	13	5 .01(10^{-1})
DD	1 2	1 2	12	1 2	12	7. 32(10^{-1})

4 Conclusions and Future Work

In this paper, we have presented DMOEA, a high performance evolutionary algorithm for multi-objective optimization problems that employs a new fitness assignment strategy and a new accepting criterion based on the principle of the minimal free energy. We have conducted experiments by computer simulation to explore the applicability of DMOEA to the complex problems of multiobjective optimization. Our computational experience shows that the proposed DMOEA can obtain very promising results. In addition, the procedure can be easily extended to solve other real-world application problems with different objectives. Therefore in the future, the applicability of DMOEA to more difficult and complex types of real-world problems, including the combinatorial optimization problems must be studied.

Acknowledgement

We would like to thank professor Kalyanmoy Deb in Indian Institute of Technology, India for warmly explaining our questions, also thank Mr. Naoki Mori in Osaka Prefecture University, Japan for provide us their valuable papers.

References:

[1] Baeck: Handbook of Evolutionary Computation. Institute of Physics Publishing, 2003.

[2] Schaffer J. D. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms, Ph. D. Thesis, Nashville, TN: Vanderbilt University, 1984.

[3] Deb Kalyanmoy: Multi-Objective Optimization using Evolutionary Algorithms, John Wiley &Sons，Chichester，U.K. 2001.

[4] Carlos A. Coello Coello, et al: Evolutionary Algorithms for Solving Multi-Objective Problems. Plenum Pub Corp, 2002.

[5] Fonseca C. M. and Fleming P. J. Genetic algorithms for multi-objective optimization: Formulation, discussion and generalization. In: S. Forrest (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms,* 1993, pp. 416–423. San Mateo, California.

[6] Horn，J.，Nafploitis N. and Goldberg D. A Niched Pareto Genetic Algorithm for Multi-objective Optimization ， In *Proceedings of the first IEEE Conference on Evolutionary Computation* ，

Michalewicz Z. Ed. Piscataway ，NJ:IEEE Press, 1994, pp.82-87.
[7] Knowles J. and Corne D. The Pareto archived evolution strategy: A new baseline algorithm for multiobjective optimization," in *Proceedings of the 1999 Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press, 1999, pp. 98–105.
[8] Srinivas N. and Deb K. Multiobjective function optimization using nondominated sorting genetic algorithms, *Evolutionary Computation*, vol. 2, no.3,1995,pp. 221–248.
[9] Deb K.，Pratap A. Agarwal S. and Meyarivan T. A Fast and Elitist Multi-objective Genetic Algorithm: NSGAⅡ，*IEEE Transaction on Evolutionary Computation* ，Vol6，No 2，2002,pp.182- 197.
[10]Zitzler E. Evolutionary algorithms for multiobjective optimization: Methods and Applications, Doctoral dissertation ETH 13398, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 1999.
[11]Zitzler E., Laumanns M. andThiele L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. TIK-Report 103, ETH Zentrum, Gloriastrasse 35, CH-8092 Zurich, Switzerland, 2001.
[12]Kita H., Yabumoto Y., Mori N. and Nishikawa Y. Multi-objective Optimization by means of Thermodynamical Genetic Algorithm，In *Proceedings of Parallel Problem Solving from Nature* Ⅳ(PPSN-Ⅳ), 1996, pp.504-512.
[13]Zitzler E., Deb K., and Thiele L. Comparison of Multi-objective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation Journal***8,** 2000, pp. 125-148.
[14]Zitzler E, Thiele L, Laumanns M et al.: Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions on Evolutionary Computation* **7**, 2003, pp.117- 132.
[15]Aarts E.H.H., Korst J.H.M, Simulated annealing and Boltzmann machines, John Wiley and Sons, 1989.
[16]Guo, T., and Kang L S.: A new evolutionary algorithm for function optimization. *Wuhan university Journal of Nature Science*. **4,** 1999, pp. 409-414.
[17]Palli,N., Azram S., McCluskey P. and Sundararajan R.: An interactive multistage ε-inequality constraint method for multiple objectives decision making. *ASME Journal of Mechanical Design* **120,** 1999, pp. 678-686.
[18]Azarm, S., B. Reynolds, and S. Narayanan, Comparison of Two Multiobjective Optimization Techniques with and within Genetic Algorithms, CD-ROM Proceedings of the 25th ASME Design Automation Conference, Paper No. DETC99/DAC-8584, Las Vegas, NV, September 1999
[19]John Eddy and Kemper Lewis. Effective Generation of Pareto Sets Using Genetic Programming. Proceedings of ASME 2001 Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Paper No. DETC2001/DAC-21094, Pittsburgh, PA, September 9-12, 2001.
[20]Kannan B.K. and Kramer S.N.: An augmented lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design. *ASME Journal of Mechanical Design* **116**, 1994, pp. 405-411.