2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Mapping Constrained Optimization Problems to Penalty Parameters: An Empirical Study

Chengyong Si, Jianqiang Shen, Xuan Zou
Shanghai-Hamburg College
University of Shanghai for Science and Technology
Shanghai 200093, China
Email: sichengyong@163.com, jianqiangshen@126.com, sherryzou@126.com

Lei Wang, Qidi Wu
College of Electronics and Information Engineering
Tongji University
Shanghai 201804, China
Email: wanglei@tongji.edu.cn, qidi@mail.tongji.edu.cn

***Abstract*—Penalty function method is one of the most popular used Constraint Handling Techniques for Evolutionary Algorithms (EAs) solution selecting, whose performance is mainly determined by penalty parameters. This paper tries to study the penalty parameter from the aspect of problem characteristics, i.e., to construct a corresponding relationship between the problems and the penalty parameters. The experimental results confirm the relationship, which provides valuable reference for future algorithm design.**

***Keywords*—constrained optimization; constraint handling techniques; differential evolution; penalty parameter; ranking methods**

I. INTRODUCTION

In the real-world applications, Constrained Optimization Problems (COPs) are very common and important. The COPs can be generally expressed by the following formulations:

$$\begin{aligned} &\text{Minimize } f(\vec{x}) \\ &\text{Subject to: } g_j(\vec{x}) \le 0, \quad j = 1, \cdots, l \\ &\qquad\qquad\quad h_j(\vec{x}) = 0, \quad j = l+1, \cdots, m \end{aligned}$$

where $\vec{x} = (x_1, \cdots, x_n)$ is the decision variable. The decision variable is bounded by the decision space S which is defined by the constraints:

$$L_i \le x_i \le U_i, \quad 1 \le i \le n \tag{1}$$

where l is the number of inequality constraints and m-l is the number of equality constraints.

The Evolutionary Algorithms (EAs) are essentially unconstraint search techniques [1] and can be mainly used to generate solutions. Equivalently, choosing the better solutions especially for the COPs is another important research area in optimization, leading to the development of various constrained optimization evolutionary algorithms (COEAs) [2]-[5]. The three most frequently used constraint handling techniques (CHTs) in COEAs are based on the concept of penalty functions, biasing feasible over infeasible solutions and multiobjective optimization.

Penalty function method is generic and applicable to any type of constraints. Its main idea is using an amount of constraint violation to punish an infeasible solution which is realized by the penalty parameters. However, the fine tuning of penalty parameters limits their real applications.

To overcome this limitation, methods were developed to separately compare the objective functions and constraint violations. For example, Deb [6] proposed a feasibility-based rule to pair-wise compare individuals:

1) Any feasible solution is preferred to any infeasible solution.

2) Among two feasible solutions, the one having better objective function value is preferred.

3) Among two infeasible solutions, the one having smaller constraint violation is preferred.

Meanwhile, multiobjective optimization technique which considers the objective function and constraint violation at the same time has been employed to handle constraints [4], [7]-[11].

Besides these basic CHTs, some other concepts like cooperative coevolution [12]-[15] and ensemble [16]-[21] have also been proposed, which can give some inspiration for designing COEAs. These methods employed different subpopulations evolving parallel. Normally, the population size of these methods changes with the evolution process. Thus, it can be seen as a dynamic adjustment process.

Among all of these aforementioned methods, the problem characteristics are rarely considered. But as Michalewicz summarized [22], it seems that Evolutionary Algorithms, in the broad sense of this term, provide just a general framework on how to approach complex problems. All their components, from the initialization, through variation operators and selection methods, to constraint-handling methods, might be problem-specific. From this, we can conclude it's essential to design a general framework which is related with the problem characteristics.

Besides, there are already some computational time complexity analyses of EAs [23]-[27] that emphasize the

This work was supported in part by the National Natural Science Foundation of China, under Grants 70871091, 61075064, 61034004, and 61005090, and in part by the Program for New Century Excellent Talents in University of Ministry of Education in China.

relationship between algorithmic features and problem characteristics which are complementary to the traditional time complexity analysis. Additionally, the importance of problem characteristics and some simple combination of algorithm variants have been realized, but the result is not so satisfactory.

For example, Tsang and Kwan [28] pointed out the need to map constraint satisfaction problems to algorithms and heuristics. But they didn't give an exact relationship between them. Mezura-Montes *et al.* [29] proposed a simple combination of two DE variants (i.e., DE/rand/1/bin and DE/best/1/bin) based on the empirical analysis of four DE variants. As only two variants are combined and the situations are not considered, the results are not competitive.

Other methods concerning the problem characteristics were also reported [30]-[31]. As presented in [30], a new method to construct the relationship between problems and algorithms as well as constraint handling techniques from the qualitative and quantitative point of view was proposed. In this paper, the problem characteristics were also summarized systematically. In [31], the authors proposed a universal model through the correspondence between problems and culture for the first time. Different algorithms can be described in this framework of this model.

Unlike the aforementioned methods, in this work, we try to study the features of penalty parameters and get the corresponding relationship between problem characteristics and penalty parameters.

The rest of this paper is organized as follows. Section II briefly introduces DE. Section III illustrates the idea of mapping problems to penalty parameters in detail. The experimental results and analysis are presented in Section IV. Finally, Section V concludes this paper and provides some possible paths for future research.

II. DIFFERENTIAL EVOLUTION (DE)

DE, which was proposed by Storn and Price [32], is a simple and efficient EA. The mutation, crossover and selection operations are introduced in DE. The first two operations are used to generate a trial vector to compete with the target vector while the third one is used to choose the better one for the next generation. To date several variants of DE have been proposed [33]. *DE/rand/1/bin* was adopted in this paper.

The population of DE consists of *NP* *n*-dimensional real-valued vectors

$$\vec{x}_i = \left\{x_{i,1}, x_{i,2}, \ldots, x_{i,n}\right\}, \quad i = 1, 2, \ldots, NP \tag{2}$$

The mutation, crossover and selection operations are defined as follows.

A. Mutation Operation

Taking into account each individual $\vec{x}_i$ (named a target vector), a mutant vector $\vec{v}_i = \left\{v_{i,1}, v_{i,2}, \ldots, v_{i,n}\right\}$ is defined as

$$\vec{v}_i = \vec{x}_{r1} + F \cdot (\vec{x}_{r2} - \vec{x}_{r3}) \tag{3}$$

where $r1$, $r2$ and $r3$ are randomly selected from [1, *NP*] and satisfying: $r1 \neq r2 \neq r3 \neq i$ and F is the scaling factor.

In this paper, if $v_{i,j}$ violates the boundary constraint, it will be reset as follows [9]:

$$v_{i,j} = \begin{cases} \min\left\{U_j, 2L_j - v_{i,j}\right\}, & if \quad v_{i,j} < L_j \\ \max\left\{L_j, 2U_j - v_{i,j}\right\}, & if \quad v_{i,j} > U_j \end{cases} \tag{4}$$

B. Crossover Operation

A trial vector $\vec{u}_i$ is generated through the binomial crossover operation on the target vector $\vec{x}_i$ and the mutant vector $\vec{v}_i$

$$u_{i,j} = \begin{cases} v_{i,j} & if\ rand_j \le C_r \quad or \quad j = j_{rand} \\ x_{i,j} & otherwise \end{cases} \tag{5}$$

where $i = 1, 2, \ldots, NP$, $j = 1, 2, \ldots, n$, j_{rand} is a randomly chosen integer within the range[1,*n*], $rand_j$ is the *j*th evaluation of a uniform random number generator within [0,1], and C_r is the crossover control parameter. The introduction of $j = j_{rand}$ can guarantee the trial vector $\vec{u}_i$ is different from its target vector $\vec{x}_i$.

C. Selection Operation

Selection operation is realized by comparing the trial vector $\vec{u}_i$ against the target vector $\vec{x}_i$ and the better one will be preserved for the next generation.

$$\vec{x}_i = \begin{cases} \vec{u}_i & if\ f(\vec{u}_i) \le f(\vec{x}_i) \\ \vec{x}_i & otherwise \end{cases} \tag{6}$$

III. MAPPING CONSTRAINED OPTIMIZATION PROBLEMS TO PENALTY PARAMETERS

A. Basic idea

As discussed in [34], the effect of the penalty parameter is directly related with the solutions, but with many repeated experiments, the result will reflect the relationship between the penalty parameters and the problem characteristics.

Based on this, the behavioral characteristics of different penalty parameter in the constraint search space are discussed, which will provide some reference for solving new constraint problems considering the problem characteristics.

The basic idea is illustrated in Fig.1, where *Pro_char_1-Pro_char_l* stands for the basic elements of the problem characteristics and *Pro_set_1-Pro_set_s* are the *s* problem sets.

By using different penalty parameters to solve different problems, the types of problems that the penalty parameters are good at solving can be obtained, and then a corresponding relationship can be constructed.

B. Penalty parameter setting

As analyzed in [34], there is no difference on ranking the population using penalty function method with a quite large penalty parameter λ (e.g., $\lambda >> \lambda_{\max}$) and a relatively smaller but still larger than $\lambda_{\max}$ penalty parameter λ (e.g.,

$\lambda = floor(\lambda_{\max}) + 1$ as they are both the same as Deb's feasibility-based rule; similarly, there is no difference on ranking the population using a quite small penalty parameter λ (e.g., $\lambda << \lambda_{\min}$) and a relatively larger but still smaller than $\lambda_{\min}$ penalty parameter λ (e.g., $\lambda = floor(\lambda_{\min}) - 1$), as they are the same as following some other rules. Here, $\lambda_{\max}$ and $\lambda_{\min}$ are determined by the current population.

Based on this conclusion, the penalty parameter in this paper is selected ranging from 0.001 to 100000. It should also be pointed out that as there is no guidelines how to set the penalty parameters in such case so as to distinguish the effect better, a simple method with the same scale (i.e., 0, 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000, 100000) is adopted.

C. Constrained problem's characteristics

Constrained optimization problem can be characterized by different parameters, e.g., equality/inequality constraints, the form of the objective function, the number of local optimal solution, etc.

In this paper, a standard for describing both from the variable and constraints as well as objective functions are adopted, which is similar as [29], but with more describing aspects. The basic four elements are: the dimension of the problem (i.e., the number of variables); the type of constraints (e.g., equality or inequality constraints); the type of objective functions (e.g., polynomial, nonlinear, linear and so on); the active constraints.

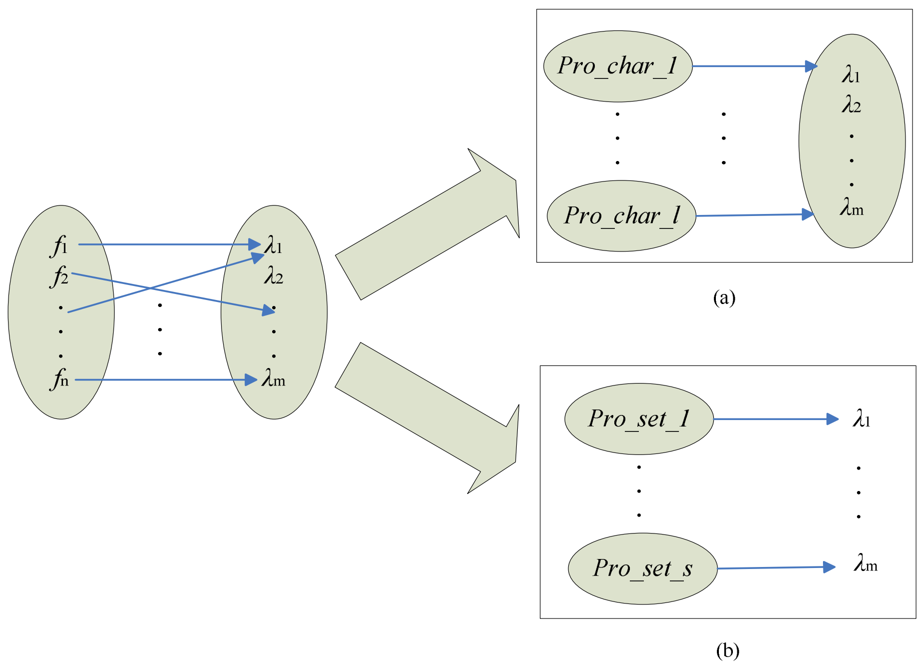

Fig.1. Illustration of the basic idea

IV. EXPERIMENTAL STUDY

A. Experimental settings

23 benchmark functions [35] were used in our experiment. The details of these benchmark functions are reported in Table I, where n is the number of decision variables, $\rho = |F|/|S|$ is the estimated ratio between the feasible region and the search space, *LI, NI, LE, NE* is the number of linear inequality constraints, nonlinear inequality constraints, linear equality constraints and nonlinear equality constraints respectively, a is the number of active constraints at the optimal solution and $f(\vec{x}^*)$ is the objective function value of the best known solution.

These benchmark functions are classified into different groups according to the problem characteristics described in Section III as shown in Table II.

The parameters in DE are set as follows: the population size (*NP*) is set to 100; the scaling factor (*F*) is randomly chosen between 0.5 and 0.6, and the crossover control parameter (*Cr*) is randomly chosen between 0.9 and 0.95.

B. Experimental results

25 independent runs were performed for each test function using 5×10^5 FES at maximum, as suggested by Liang *et al.* [35]. Additionally, the tolerance value δ for the equality constraints was set to 0.0001.

Table III-IV shows the best value and SR (succeed rate) for different penalty parameters on different problems. Here, "-"in Table III means not available, i.e., there is no feasible solution in 25 independent runs.

From Table III-IV, it can be observed that the results by different penalty parameters vary considerably.

TABLE I. DETAILS OF THE BENCHMARK FUNCTIONS

Problem	***n***	**Type of objective function**	***ρ***	***LI***	***NI***	***LE***	***NE***	***a***	$f(\vec{x}^*)$
g01	13	quadratic	0.0111%	9	0	0	0	6	-15.0000000000
g02	20	nonlinear	99.9971%	0	2	0	0	1	-0.8036191042
g03	10	polynomial	0.0000%	0	0	0	1	1	-1.0005001000
g04	5	quadratic	52.1230%	0	6	0	0	2	-30665.5386717834
g05	4	cubic	0.0000%	2	0	0	3	3	5126.4967140071
g06	2	cubic	0.0066%	0	2	0	0	2	-6961.8138755802
g07	10	quadratic	0.0003%	3	5	0	0	6	24.3062090681
g08	2	nonlinear	0.8560%	0	2	0	0	0	-0.0958250415
g09	7	polynomial	0.5121%	0	4	0	0	2	680.6300573745
g10	8	linear	0.0010%	3	3	0	0	6	7049.2480205286
g11	2	quadratic	0.0000%	0	0	0	1	1	-0.7499000000
g12	3	quadratic	4.7713%	0	1	0	0	0	-1.0000000000
g13	5	nonlinear	0.0000%	0	0	0	3	3	0.0539415140
g14	10	nonlinear	0.0000%	0	0	3	0	3	-47.7648884595
g15	3	quadratic	0.0000%	0	0	1	1	2	961.7150222899
g16	5	nonlinear	0.0204%	4	34	0	0	4	-1.9051552586
g17	6	nonlinear	0.0000%	0	0	0	4	4	8853.5396748065
g18	9	quadratic	0.0000%	0	13	0	0	6	-0.8660254038
g19	15	nonlinear	33.4761%	0	5	0	0	0	32.6555929502
g20	24	linear	0.0000%	0	6	2	12	16	0.2049794002
g21	7	linear	0.0000%	0	1	0	5	6	193.7245100700
g22	22	linear	0.0000%	0	1	8	11	19	236.4309755040
g23	9	linear	0.0000%	0	2	3	1	6	-400.0551000000
g24	2	linear	79.6556%	0	2	0	0	2	-5.5080132716

TABLE II. CLASSIFICATION OF BENCHMARK FUNCTIONS

Problem characteristics		**Problems**
Number of variables	10-20 (High)	g01, g02, g03, g07, g14, g19
	5-9 (Medium)	g04, g09, g10, g13, g16, g17, g18, g21, g23
	2-4 (Low)	g05, g06, g08, g11, g12, g15, g24
Type of constraints	Only inequalities	g01, g02, g04, g06, g07, g08, g09, g10, g12, g16, g18, g19, g24
	Only equalities	g03, g11, g13, g14, g15, g17
	Both inequalities and equalities	g05, g21, g23
Type of objective function	Polynomial (including quadratic and cubic)	g01, g03, g04, g05, g06, g07, g09, g11, g12, g15, g18
	Nonlinear	g02, g08, g13, g14, g16, g17, g19
	Linear	g10, g21, g23, g24
Type of active constraints	$a<N$	g01, g02, g04, g05, g07, g08, g09, g12, g16, g18, g19
	$a=N$	g03, g06, g10, g11, g13, g14, g15, g17, g21, g23, g24

TABLE III. BEST VALUE FOR DIFFERENT PENALTY PARAMETERS

λ / Pro.	0	0.001	0.01	0.1	1	10	100	1000	10000	100000
g01	—	—	—	—	-15	-15	-15	-15	-15	-15
g02	-0.6858	-0.6396	-0.6427	-0.6495	-0.803619	-0.803619	-0.803619	-0.803619	-0.803619	-0.803619
g03	—	—	—	—	—	—	—	-1.0005	-0.9939	-1.0005
g04	-30420.114	-30469.626	-30344.143	-30527.218	-30510.039	-30506.204	-30482.027	-30665.538	-30665.538	-30665.538
g05	—	—	—	—	—	5126.4967	5126.4967	5126.4967	5126.4967	5126.4967
g06	6882.9428	6866.9363	6509.7585	6759.1629	6870.7776	-6873.0142	-6901.9516	-6948.0697	-6961.8139	-6961.8139
g07	68.7315	40.8588	71.0100	55.2424	25.8322	24.3062	24.3062	24.3062	24.3062	24.3062
g08	-0.095805	-0.095825	-0.095820	-0.095824	-0.095823	-0.095824	-0.095825	-0.095825	-0.095825	-0.095825
g09	685.3741	703.7783	690.5985	684.1232	680.7736	680.6300	680.6300	680.6300	680.6300	680.6300
g10	—	—	—	—	—	—	—	—	7049.2480	7049.2480
g11	—	0.8188	0.7608	0.9402	0.7499	0.7499	0.7499	0.7499	0.7499	0.7499
g12	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
g13	—	—	—	0.0539415	0.0539415	0.3402	0.7025	0.9621	0.8766	0.6992
g14	—	—	—	—	—	—	-47.764888	-47.764888	-47.764888	-47.764888
g15	—	—	—	—	—	961.715022	961.715022	961.715022	961.715022	961.715022
g16	-1.4667	-1.5400	-1.7355	-1.9050	-1.9051	-1.9051	-1.9051	-1.9051	-1.9051	-1.9051
g17	—	—	—	—	—		8853.5339	8860.2247	8862.5701	8871.5957
g18	—	—	—	—	-0.866025	-0.866025	-0.866025	-0.866025	-0.866025	-0.866025
g19	51.2803	45.3600	47.2818	42.2332	32.6559	32.6559	32.6559	32.6559	32.6559	32.6559
g21	—	—	—	—	—	—	—	193.7280	193.7658	193.7732
g23	—	—	—	—	—	—	—	-400.0551	-399.7722	-384.6801
g24	-5.4878	-5.4903	-5.4934	-5.4927	-5.5080	-5.5080	-5.5080	-5.5080	-5.5080	-5.5080

TABLE IV. SUCCESS RATE FOR DIFFERENT PENALTY PARAMETERS

λ / Pro.	0	0.001	0.01	0.1	1	10	100	1000	10000	100000
g01	0	0	0	0	1	1	1	1	1	1
g02	0	0	0	0	0.92	0.88	0.92	0.72	0.76	0.80
g03	0	0	0	0	0	0	0	**0.28**	0	0.04
g04	0	0	0	0	0	0	0	1	1	1
g05	0	0	0	0	0	**1**	**1**	**0.24**	0	0
g06	0	0	0	0	0	0	0	0	**0.56**	0
g07	0	0	0	0	0	1	1	1	1	1
g08	0.16	0.16	0.12	0.40	0.32	0.68	1	1	1	1
g09	0	0	0	0	0	1	1	1	1	1
g10	0	0	0	0	0	0	0	0	**0.32**	0
g11	0	0	0	0	1	1	1	1	0.60	0.68
g12	1	1	1	1	1	1	1	1	1	1
g13	0	0	0	**1**	0.20	0	0	0	0	0
g14	0	0	0	0	0	0	1	0.96	0.48	0.48
g15	0	0	0	0	0	1	0.36	0.16	0.04	0.20
g16	0	0	0	0	1	1	1	1	1	1
g17	0	0	0	0	0	—	**1**	0	0	0
g18	0	0	0	0	1	1	1	1	1	1
g19	0	0	0	0	1	1	1	1	1	1
g21	0	0	0	0	0	0	0	0	0	0
g23	0	0	0	0	0	0	0	**1**	0	0
g24	0	0	0	0	1	1	1	1	1	1

TABLE V. CORRESPONDING RELATIONSHIP BETWEEN PENALTY PARAMETER AND PROBLEMS

Result	λ	**Problem**	**Problem characteristics**
Always	any	g08、g12	Low dimension, *NI*, *a*<*N*
Continuity	$\lambda \geq 1$	g01、g16 、g18、g19、g24、g02、g11	High/Median dimension, *LI*/*NI*, not linear, *a*<*N* or low dimension, *NI*, linear, *a*=*N*
	$\lambda \geq 10$	g07、g09	High/Median dimension, *NI*, polynomial, *a*<*N*
	$\lambda \geq 100$	g14	High dimension, *LE*, nonlinear, *a*=*N*
	$\lambda \geq 1000$	g04	Median dimension, *NI*, *a*<*N*
Intermittent	0.1	g13	Median dimension, *NE*, nonlinear_1, *a*=*N*
	10	g15	Low dimension, *LE*+*NE*, *a*=*N*
	100	g17	Median dimension, *NE*, nonlinear_2, *a*=*N*
	1000	g03、g23	High/Median dimension, *NE*, *a*=*N*
	10000	g06、g10	High/Low dimension, *LI*/*NI*, *a*=*N*
	10/100/1000	g05	Low dimension, *LI*+*NE*, *a*<*N*

C. Corresponding relationship

This section is mainly from the micro-level to determine the specific corresponding relationship, i.e., first find the problem set (*Pro_set_1*-*Pro_set_s*) for each penalty parameter λ, and then summarize the problem characteristics for each λ that they are suited to solve from the perspective of different combination of basic elements. A basic principle of this part is to find the unique features that other problems do not have.

Also, it's important to identify some of the problems that can reach the best value for certain penalty parameter value λ or after certain value of λ, which will provide useful conclusion.

To sum up, the corresponding relationship between constrained optimization problems and penalty parameters can be listed in Table V. Here, in the column of "result", "Always" means that the penalty parameters will not affect the results; "Continuity" means that when penalty parameter is lager than some value, it can always find the optimal value; "Intermittent" means that the optimal value can be obtained only with certain penalty parameter values.

Considering the importance of the type of constraints, they are classified into nine types in this section, i.e., *LI*, *NI*, *LI+NI*, *LE*, *NE*, *LE+NE*, *LI+NE*, *NI+NE*, *NI+LE+NE*. In accordance with the concept of permutations and combinations, there should be 16 different combinations for these four elements, which is also one of the limitations for this paper.

Besides, for functions g13 and g17, though the type of objective functions are both nonlinear, the objective function of g13 is exponential, while that of g17 is the summarization of two piecewise linear functions, so they are labeled as nonlinear_1 and nonlinear_2 respectively in Table V.

From the analysis, we can conclude that the element of problem characteristics plays an important role in getting a reasonable corresponding relationship, which will also be the future work.

V. CONCLUSION

This paper has presented a new way to get the corresponding relationship between constrained optimization problems and the penalty parameters in penalty function methods. 23 benchmark functions with typical problem characteristics collected in the IEEE CEC2006 special session on constrained real-parameter optimization were utilized to verify the corresponding relationship.

The relationship obtained reflects some characteristics of penalty parameters to some extent, but the inner mechanisms of penalty parameters should be theoretically studied. Also, whether the relationship is related with the evolutionary algorithms is to be verified in future research.

The problem characteristics summarized in this paper are based on the benchmark functions, but as Z. Michalewicz concluded [22], there is no comparison in terms of complexity between real-world problems and toy problems, and real-world applications usually require hybrid approaches where an 'evolutionary algorithm' is loaded with non-standard features (e.g., decoders, problem-specific variation operators, memory), so how to apply these conclusions to the real-world problems is still a challenging work and will be our future work.

ACKNOWLEDGMENT

Chengyong Si would like to thank Prof. Dr. Robert Weigel for his great help in the life and research work, as this work is partially done when he was with the Institute for Electronics Engineering, University of Erlangen-Nuernberg in Germany as a joint doctor, and he is grateful to Prof. Xin Yao for the valuable suggestions. The authors also gratefully acknowledge Dr. T. Lan and Dr. J. Hu for improving the presentation of this paper.

REFERENCES

[1] E. Mezura-Montes and C. A. Coello Coello, “A simple multimembered evolution strategy to solve constrained optimization problems,” *IEEE Trans. Evol. Comput.*, vol. 9, no. 1, pp. 1–17, Feb. 2005.

[2] Z. Michalewicz and M. Schoenauer, “Evolutionary algorithm for constrained parameter optimization problems,” *Evol. Comput.*, vol. 4, no. 1, pp. 1–32, Feb. 1996.

[3] C. A. Coello Coello, “Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art,” *Comput. Methods Appl. Mech. Eng.*, vol. 191, no. 11/12, pp. 1245–1287, Jan. 2002.

[4] Z. Cai and Y. Wang, “A multiobjective optimization-based evolutionary algorithm for constrained optimization,” *IEEE Trans. Evol. Comput.*, vol. 10, no. 6, pp. 658–675, Dec. 2006.

[5] E. Mezura-Montes and C. A. Coello Coello, “Constraint-handling in nature-inspired numerical optimization: Past, present and future,” *Swarm Evol. Comput.*, Vol. 1, No. 4, pp. 173-194, Dec. 2011.

[6] K. Deb, “An efficient constraint handling method for genetic algorithms,” *Comput. Methods Appl. Mech. Eng.*, vol. 186, no. 2–4, pp. 311–338, Jun. 2000.

[7] Y. Wang, Z. Cai, G. Guo, and Y. Zhou, “Multiobjective optimization and hybrid evolutionary algorithm to solve constrained optimization problems,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 3, pp. 560–575, Jun. 2007.

[8] A. Zhou, B-Y. Qu, H. Li, S-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state-of-the-art", *Swarm Evol. Comput.*, Vol. 1, No. 1, pp. 32-49, Mar. 2011.

[9] A. Menchaca-Mendez and C. A. Coello Coello, “Solving multiobjective optimization problems using differential evolution and a maximin selection criterion”, in *proc. CEC*, 2012, pp. 3143-3150.

[10] Y. Wang and Z. Cai , “Combining multiobjective optimization with defferential evolution to solve constrained optimization problems” *IEEE Trans. Evol. Comput.*, vol. 16, no. 1, pp. 117–134, Feb. 2012.

[11] A. Ponsich, A. L. Jaimes, and C. A. Coello Coello, “A survey on multiobjective evolutionary algorithms for the solutions of the portfolio optimization problems and other finance and economics applications,” *IEEE Trans. Evol. Comput.*, 2012 (accepted). DOI: 10.1109/ TEVC.2012.2196800.

[12] F. V. den Bergh and A. Engelbrecht, “A cooperative approach to particle swarm optimization,” *IEEE Tran. Evol. Comput.*, Vol. 8, no. 3, pp. 225-239, Jun. 2004.

[13] Z. Yang, K. Tang, and X. Yao, “Large scale evolutionary optimization using cooperative coevolution,” *Inform. Sci.*, Vol. 178, pp. 2986-2999, Aug. 2008.

[14] X. Li and X. Yao, “Cooperatively coevolving particle swarms for large scale optimization,” *IEEE Tran. Evol. Comput.*, Vol. 16, no. 2, pp. 210-224, Apr. 2012.

[15] F. Peng, K. Tang, G. Chen, and X. Yao, “Population-based algorithm portfolios for numerical optimization,” *IEEE Tran. Evol. Comput.*, Vol. 14, no. 5, pp. 782-800, Oct. 2010.

[16] R. Mallipeddi, S. Mallipeddi, and P. N. Suganthan, “Ensemble strategies with adaptive evolutionary programming,” *Inform. Sci.*, Vol. 180, pp. 1571-1581, Jan. 2010.

[17] Q. Pan, P. N. Suganthan, and M. F. Tasgetiren, “A harmony search algorithm with ensemble of parameter sets,” in *Proc. CEC*, 2009, pp. 1815-1820.

[18] S. Z. Zhao, P. N. Suganthan, and Q. Zhang, "Decomposition based multiobjective evolutionary algorithm with an ensemble of neighborhood sizes", *IEEE Trans. Evol. Comput.*, Vol. 16, No. 3, pp. 442-446, Jun. 2012.

[19] E.L.Yu and P. N. Suganthan, “Ensemble of niching algorithms,” *Inform. Sci.*, Vol. 180, pp. 2815-2833, Apr. 2010.

[20] R. Mallipeddi and P. N. Suganthan, “Ensemble of constraint handling techniques,” *IEEE Tran. Evol. Comput.*, Vol. 14, no. 4, pp. 561-579, Aug. 2010.

[21] Y. Wang, Z. Cai, and Q. Zhang, “Differential evolution with composite trial vector generation strategies and control parameters,” *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 55–66, Feb. 2011.

[22] Z. Machalewicz, “Quo Vadis, Evolutionary Computation? On a growing gap between theory and practice,” in *WCCI 2012 Plenary/Invited Lectures*, J. Liu et al., Eds. LNCS7311, pp. 98-121.

[23] J. He and X. Yao, “Drift analysis and average time complexity of evolutionary algorithms,” *Artif. Intell.*, vol. 127, no. 1, pp. 57–85, Mar. 2001.

[24] J. He and X. Yao, “A study of drift analysis for estimating computation time of evolutionary algorithms,” *Nat. Comput.*, vol. 3, no. 1, pp. 21–35, Mar. 2004.

[25] T. Chen, K. Tang, G. Chen and X. Yao, “Analysis of computational time of simple estimation of distribution algorithms,” *IEEE Trans. Evol. Comput.*, vol. 14, no. 1, pp.1–22, Feb. 2010.

[26] P. Lehre and X. Yao, “Runtime analysis of the (1+1) EA on computing unique input output sequences,” *Inform. Sci.*, in press. DOI: 10.1016/j.ins.2010.01.031.

[27] X. Yao, “Unpacking and understand Evolutionary Algorithms,” in *WCCI 2012 Plenary/Invited Lectures*, J. Liu et al., Eds. LNCS7311, pp. 60-76.

[28] E. Tsang and A. Kwan, “Mapping constraint satisfaction problems to algorithms and heuristics,” Dept. Comput. Sci., Univ. Essex, Colchester, UK, Tech. Rep. CSM-198, 1993.

[29] E. Mezura-Montes, M. E. Miranda-Varela, and R. C. Gómez-Ramón, “Differential evolution in constrained numerical optimization: an empirical study,” *Inform. Sci.*, vol.180, no.22, pp. 4223-4262, Nov. 2010.

[30] C. Si, L.Wang, and Q. Wu, “Mapping constrained optimization problems to algorithms and constraint handling techniques”, in *proc. CEC*, 2012, pp. 3308-3315.

[31] C. Si, H. Tian, L.Wang, and Q. Wu, “A generalized culture-based optimization model”, in *Chinese Control Conference*, 2012, pp. 7195-7198.

[32] R. Storn and K. Price, “Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces,” Int. Comput. Sci. Instit., Berkeley, CA, Tech. Rep. TR-95-012, 1995.

[33] S. Das and P. N. Suganthan, "Differential evolution: A Survey of the state-of-the-art", *IEEE Trans. Evol. Comput.*, Vol. 15, No. 1, pp. 4-31, Feb. 2011.

[34] C. Si, T. Lan, J. Hu, L.Wang, and Q. Wu, “On the penalty parameter of the penalty function method”, *Control and Decision,* to be published.

[35] J. J. Liang, T. P. Runarsson, E. Mezura-Montes, M. Clerc, P. N. Suganthan, C. A. Coello Coello, and K. Deb, “Problem definitions and evaluation criteria for the CEC 2006,” *Special Session on Constrained Real-Parameter Optimization*, Nanyang Technol. Univ., Singapore, Tech. Rep., 2006.