Differential Evolution Algorithm with PCA-based Crossover

Yuan-long Li, Wei-neng Chen and Jun Zhang (Corresponding Author)
Department of Computer Science, Sun Yat-sen University, P.R. China
Key Laboratory of Digital Life, Ministry of Education, P.R. China
Key Laboratory of Software Technology, Education Dept. of Guangdong Province, P.R. China
junzhang@ieee.org

ABSTRACT
Crossover is a very important operation in current differential evolution (DE) algorithms. The existing crossover strategies in DE show promising effects especially when the algorithms are applied to separable functions. However, the operation fails to work well when applied to the ill-conditioned and inseparable problems because the recombination of good genes is no longer promising for generating better individuals. In this paper, we propose to use the principal component analysis (PCA) technique to rebuild a coordinate system. With this system, the correlations among variables are decreased for the crossover operation of DE and the crossover operation become more efficient.

Categories and Subject Descriptors
B.2.4 [**Arithmetic and Logic Structures**]: High-Speed Arithmetic –*Algorithms, Cost/performance.*

General Terms
Algorithms.

Keywords
Differential Evolution; principle component analysis; CMA-ES; crossover;

1. INTRODUCTION
Differential evolution (DE) algorithm has been one of the most popular evolutionary continuous optimization methods since its first publication [1] [2]. Like other continuous optimization methods such as particle swarm optimization (PSO) [7][9][10][11] algorithm, the DE algorithm uses real-coded genes and linear combination searching strategy. Such strategy is shown to be more effective for the continuous optimization problems than the original binary coding scheme and random mutation based search method in genetic algorithm (GA) [8]. DE is different from PSO on that it applies the crossover operation of GA. The crossover operation in DE is still important because it shows good performance on the separable functions. Currently the most popular online method to solve the correlation problem in evolution computing was proposed by covariance matrix adaptation-evolution strategy (CMA-ES) [4]. The coordinate system built in CMA-ES by the eigenvectors of the covariance matrix of the population is actually the coordinate system used in principal component analysis (PCA) [5]. Here we propose the PCA-based crossover operation for DE.

GECCO'12 Companion, July 7–11, 2012, Philadelphia, PA, USA.
ACM 978-1-4503-1178-6/12/07.

2. PCA-based Crossover for DE Algorithm
The new coordinate system rebuilt by PCA is useful for the crossover operation to work theoretically. To do the crossover in the new coordinate system, the generated temporary mutated and the parent population should be rotated into the new coordinate system to do the crossover and then the newly generated population should be rotated back to be evaluated. Suppose the mutated temporary population is indicated by a matrix V and the parents' population by matrix P, with their rows standing for individuals. The mean vector of P is put into a matrix M with its each row set to be the mean vector. Then the rotation from the original coordinate system to the new coordinate system can be done by (1) and (2):

$$V_r = (V - M)B^{-1}, \tag{1}$$

$$P_r = (P - M)B^{-1}, \tag{2}$$

where B is the eigenvectors matrix and D is the eigenvalues diagonal matrix of the covariance matrix of the current population. After the crossover operation, temporary population in the new coordinate system U_r is generated from the recombination of V_r and P_r. Then U_r will be rotated back to the original system by (8).

$$U = U_r B + M \tag{3}$$

Equation (3) gives the donor population U in the original coordinate system which will be further compared with the original population in the selection process of DE.

During the evolution process, the rotation matrix B should be updated every K generations. K is not necessary to be 1 (i.e., update B at every generation). Usually K can be set larger because the distribution of the population evolves eventually and no sudden great change will happen.

3. Experiments
To test the PCA-based crossover for DE, we use one of the best and simplest DE variants JADE [3] as the test DE algorithm. All the original parameters of JADE are kept unchanged [4] while the new parameters caused by PCA-based crossover are set as follows. The number of generations K to do the updating of B is set to different values 30, 40 and 50 to test its affects to the performance of the algorithm. The JADE/PCACR is compared in Table 1 with the original JADE algorithm on the 30-dimensional CEC'05 test suite [6]. The stop condition is set to end when the maximum number of function evaluations (NFEs) is encountered. The maximum NFEs is set to be 3×10^5 for most of the test functions.

For test function 1 and 9, the maximum NFEs are set to 5×10^4 and 1×10^5.

4. Conclusion

In this paper, the PCA-based crossover is tested for DE algorithm. As the crossover operation in DE requires the variables to be as separable as possible, the PCA-based coordinate system rotation can be useful to decompose the correlations among the original variables.

5. REFERENCES

[1] R. Storn and K. V. Price, "Differential evolution–A simple and efficient adaptive scheme for global optimization over continuous spaces," Institute of Company Secretaries of India, Chennai, Tamil Nadu. Tech. Report TR-95-012, 1995.

[2] R. Storn and K. V. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," J. Global Optimization, vol. 11, no. 4, pp. 341–359, 1997.

[3] Jingqiao Zhang, and Arthur C. Sanderson, "JADE: adaptive differential evolution with optional external archive", IEEE Trans. Evol. Comput., vol. 13,no. 5,pp. 945–958, Jun. 2009.

[4] A. Auger and N. Hansen, A restart CMA evolution strategy with increasing population size, 2005.

[5] K. Pearson, "On Lines and Planes of Closest Fit to Systems of Points in Space". Philosophical Magazine 2 (6), 1901,pp. 559–572.

[6] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A. Auger,and S. Tiwari, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," Nanyang Technol. Univ., Singapore, and IIT Kanpur, Kanpur, India, KanGAL Rep. #2005005, May 2005.

[7] Jun Zhang, Zhi-hui Zhan, Ying Lin, Ni Chen, Yue-jiao Gong, Henry S.H. Chung, Yun Li and Yu-hui Shi, "Evolutionary Computation Meets Machine Learning: A Survey", IEEE Computational Intelligence Magazine, pp.68-75, NOVEMBER 2011.

[8] Jun Zhang, Henry Chung and W.L., LO, "Clustering-Based Adaptive Crossover and Mutation Probabilities for Genetic Algorithms", IEEE Transactions on Evolutionary Computation Vol.11, No.3, June 2007, Page. 326-335.

[9] W.N. Chen, Jun Zhang, Henry Chung, W.L., Zhong, W.G. Wu and Y.H., Shi, "A Novel Set-Based Particle Swarm Optimization Method for Discrete Optimization Problems", IEEE Transactions on Evolutionary Computation, Vol.14,No.2, pp.278-300, April 2010.

[10] Z.H. Zhan, Jun Zhang, Y. Li and Henry Chung, "Adaptive Particle Swarm Optimization", IEEE Transactions on Systems, Man, and Cybernetics--Part B. VOL. 39, NO. 6, Dec 2009, Page 1362-1381.

[11] W.N. Chen and Jun Zhang, "Ant Colony Optimization Approach to Grid Workflow Scheduling Problem with Various QoS Requirements", IEEE Transactions on Systems, Man, and Cybernetics--Part C: Applications and Reviews, Vol. 31, No. 1,pp.29-43,Jan 2009.

Table 1. JADE/PCACR compared to the original JADE with *K*=30, 40 and 50 on 30-dimensional CEC'05 test suite. The mean fitness values and standard deviations of 50 independent runs are shown. Better mean fitness values are shown in bold type.

D=30	JADE		JADE_PCACR/K=30		JADE_PCACR/K=40		JADE_PCACR/K=50	
F(x)	mean	std	mean	std	mean	std	mean	std
1	1.15E-15	4.54E-15	**9.29E-16**	1.93E-15	**5.85E-16**	1.38E-15	**6.56E-16**	1.36E-15
2	2.65E-28	1.31E-28	**8.97E-29**	4.76E-29	**8.52E-29**	5.74E-29	**6.73E-29**	4.81E-29
3	6.68E+03	4.03E+03	**2.89E-22**	6.28E-22	**1.47E-22**	3.58E-22	**8.08E-23**	7.23E-23
4	1.44E-16	4.94E-16	2.82E-13	1.84E-12	**2.47E-18**	1.62E-17	**2.80E-23**	1.98E-22
5	2.86E-07	1.39E-06	1.52E+01	5.02E+01	7.62E-04	5.37E-03	3.56E+01	1.89E+02
6	7.13E+00	2.52E+01	**4.93E+00**	2.91E+01	**7.55E-01**	1.92E+00	**8.95E-01**	2.69E+00
7	8.38E-03	6.07E-03	**3.70E-03**	5.64E-03	**2.96E-03**	5.38E-03	**3.94E-03**	6.21E-03
8	2.09E+01	1.74E-01	2.09E+01	1.06E-01	2.09E+01	6.79E-02	2.09E+01	1.43E-01
9	5.99E-05	3.58E-05	4.98E+01	7.50E+00	4.76E+01	6.53E+00	4.80E+01	8.21E+00
10	2.54E+01	5.02E+00	2.65E+01	6.24E+00	**2.49E+01**	5.55E+00	2.63E+01	4.99E+00
11	2.51E+01	1.66E+00	2.55E+01	1.43E+00	2.54E+01	1.60E+00	2.55E+01	1.72E+00
12	6.37E+03	4.39E+03	**2.40E+03**	3.83E+03	**3.71E+03**	5.33E+03	**3.37E+03**	3.69E+03
13	1.48E+00	1.03E-01	4.09E+00	4.56E-01	4.06E+00	4.87E-01	3.97E+00	5.26E-01
14	1.22E+01	3.69E-01	**1.21E+01**	3.35E-01	**1.21E+01**	3.74E-01	**1.22E+01**	3.07E-01
15	3.40E+02	1.05E+02	3.84E+02	9.07E+01	**3.22E+02**	9.20E+01	3.72E+02	8.33E+01
16	8.85E+01	1.21E+02	9.48E+01	1.07E+02	9.52E+01	1.05E+02	1.15E+02	1.40E+02
17	1.10E+02	1.14E+02	1.81E+02	1.71E+02	1.28E+02	9.81E+01	1.69E+02	1.54E+02
18	9.044E+02	1.03E+00	**8.94E+02**	7.40E+00	**8.94E+02**	6.12E+00	**8.94E+02**	5.31E+00