http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (1): 105-112, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.105.112

Influence of Dimensionality and Population Size on Opposition-based Differential Evolution Using the Current Optimum

[1]Qingzheng Xu, [1]Na Wang and [2]Rong Fei
[1]Xi'an Communication Institute, Xi'an 710106, China
[2]School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China

Abstract: Diverse forms of opposition are already existent virtually everywhere around us and utilizing opposite numbers to accelerate an optimization method is a new idea. In this study, three algorithms (DE, COODE and IODE) are compared for different problem dimensions and different population sizes. Experiments on 58 widely used benchmark problems show that, opposition-based differential evolution using the current optimum performs better than the original algorithm for larger population size which is usually required for more complex and high-dimensional problems.

Key words: Opposition-based learning, dimensionality, population size, opposite point, function optimization

INTRODUCTION

Diverse forms of opposition are already existent virtually everywhere around us but the nature and significance of oppositeness is well understood only in specific contexts within the fields of philosophy, linguistics, psychology, logic and physics. The interplay between entities and opposite entities is apparently fundamental for maintaining universal balance (Tizhoosh *et al*., 2008).

The basic concept of Opposition-based Learning (OBL) was original introduced by Tizhoosh (2005) and was successfully applied to several problems, such as large scale optimization problem (Rahnamayan and Wang, 2008; Wang *et al*., 2011), optimization of noisy problem (Han and He, 2007; Rahnamayan *et al*., 2006), multi-objective optimization (Dong and Wang, 2009; Peng *et al*., 2008), traveling salesman problem (Malisia and Tizhoosh, 2007), data mining (Kanzawa *et al*., 2007; Rashid and Baig, 2010), nonlinear system identification (Subudhi and Jena, 2011), etc. The main idea of this optimization is to estimate a certain individual and its corresponding opposite individual at the same time, in order to achieve a better approximation for the candidate solution. A mathematical proof was proposed by Rahnamayan *et al*. (2008a) to show that, in general, opposite candidate solutions are more likely to be closer to the global optimal solution than a purely random one. During population evolution, the symmetry point between a candidate solution and its corresponding opposite candidate solution is changing dramatically with the geometric center of variables's initial range or current interval. The optimization algorithm using opposition-based learning strategy may explore infertile filed when the global optimum keeps away from the center of variables's interval boundaries, especially in the later stage of evolution. Obviously, it can result in the lower utilization rate of opposite numbers and the poorer algorithm performance for consuming a lot of evaluation opportunities inefficiency.

A novel opposition-based learning strategy using the current optimum (COOBL) was first proposed and was initially combined with differential evolution for function optimization (Xu *et al*., 2011a, b). The optimum in the current generation was served as a symmetry point between a candidate solution and the corresponding opposite candidate solution, resulting in a high rate of opposite population usage during the later stage of evolution. Experiments results clearly show that, the proposed algorithm, opposition-based differential evolution using the current optimum (COODE), is capable of improving performance significantly because of opposite points. The contribution of opposite points using the current optimum to the acceleration process was confirmed by the further results and was not reproducible by additional random sampling. Additionally, an improved version of opposition-based differential evolution (IODE) is proposed to reveal ideal and perfect results using opposition-based learning.

The performance of COODE algorithm highly depends on the control parameters function dimension and population size. Different parameter settings will lead to different performances. In this study, the effects of function dimension and population size on the speedup of COODE are investigated completely.

Corresponding Author: Qingzheng Xu, Xi'an Communication Institute, Xi'an 710106, China, Zhangba East Road, Xi'an 710065, China

INFLUENCE OF DIMENSIONALITY AND POPULATION SIZE ON COODE

Experimental setup: To study the influence of dimensionality and population size, DE (differential evolution), COODE and IODE are compared for different problem dimensions and different population sizes. The pseudo-code of three algorithms and more details can be found (Xu *et al*., 2011a, b). However, it should be noted that the results obtained by IODE are just for reference and are not feasible for optimization problem because that the global optimum of functions is unknown in advance.

A comprehensive set of numerical benchmark functions, including 58 different well-known global optimization problems (Rahnamayan *et al*., 2008b), has been employed for performance verification. The utilized test suite includes unimodal as well as highly multimodal minimization problems. The dimensionality of problems varies from 2-30 to cover a wide range of problem complexity.

For all conducted experiments, parameter settings are as follows. These values have been chosen according to reported setting in the previous literature (Rahnamayan *et al*., 2008b) and so there has no new attempts to obtain better values for them. The termination criterion is to find a value smaller than the Value-to-reach (VTR) before reaching the maximum Number of Function Calls (NFC):

- Population size, $N_p = 100$
- Differential amplification factor, $F = 0.5$
- Crossover probability constant, $C_r = 0.9$
- Jumping rate constant, $J_r = 0.3$
- Mutation strategy: DE/rand/1/bin (classic version of DE)
- Maximum NFC, $MAX_{NFC} = 10^6$
- Value to reach, $VTR = 10^{-8}$

The convergence speed is compared by measuring the number of function calls which is the most commonly used metric in literatures. A smaller NFC means higher convergence speed. In order to minimize the effect of the stochastic nature of the algorithms on the measured metric, the reported NFC for each function is the average over 100 trials.

In order to compare convergence speeds, the Acceleration Rate (AR) is used which is defined as follows, based on the NFC for the two algorithms A and B:

$$AR_{A/B} = \begin{cases} -\frac{NFC_A}{NFC_B}, & \text{if } NFC_A > NFC_B \\ \frac{NFC_B}{NFC_A}, & \text{else} \end{cases} \quad (1)$$

Obviously, when $AR_{A/B}>1$, the convergence speed of algorithm A is superior to that of algorithm B. When $AR_{A/B}<-1$, algorithm B runs faster than algorithm A.

The number of times, for which the algorithm successfully reaches the VTR for each test function, is measured as the Success Rate (SR):

$$SR = \frac{\text{No. of times reached VTR}}{\text{Total No. of trials}} \quad (2)$$

Experiment series 1: Influence of dimensionality: In order to investigate the effect of the problem dimensionality, the same experiments are repeated for D/2 and 2D for each scalable function from the test set. All other control parameters are kept unchanged. Results for D/2 and 2D are given in Table 1 for 42 test functions. For problems f_{18}, f_{51} and f_{52}, the global optimum and its location change quickly for the varying dimension of the problems. The data are analyzed with software Statistical Product and Service Solutions (SPSS) 14.0 version and significance is assumed at 0.05. The better results of the NFC and the SR for different dimension sizes are highlighted in boldface.

According to the obtained results, except for function f_{23}(D = 60), on which there is no significant difference in the statistical sense, COODE surpasses DE on 31 test functions while DE outperforms COODE on three functions (f_4(D = 60), f_5(D = 20) and f_{18}(D = 20)). Both algorithms are unable to solve problems f_{24}, f_{51} and f_{56} for D/2 and 2D before meeting the maximum NFC. The average AR in three dimensions is equal to -1.57, meaning that COODE performs 57% faster than DE. The average SR for DE and COODE are 0.73 and 0.67, respectively. Now, we can conclude that COODE has demonstrated outstanding performance on NFC again.

For 12 functions (f_1, f_2, f_3, f_6, f_7, f_8, f_{15}, f_{19}, f_{21}, f_{30}, f_{41} and f_{52}), the AR is increased by growing dimensionality while AR is decreased for four functions (f_4, f_5, f_{18} and f_{31}). The definition of AR tells us that, the greater AR means the better convergence speed of the algorithm. Then we can conclude easily that, to solve high-dimensional problems, COODE provides a more clear advantage in convergence speed when compared with DE algorithm. An interesting effect for f_{23} is that for dimensions 15 and 30, COODE performs better than DE; but when the dimension is increased to 60, DE shows better results in terms of NFC. Furthermore, COODE cannot solve f_4 for D = 60, f_5 for

Table 1: Comparison of DE, COODE and IODE for different dimension sizes

		DE							IODE			
		NFC			SR		COODE		NFC		SR	
Function	D	Avg	Sig.	AR	Avg	Sig.	NFC	SR	Avg	Sig.	Avg	Sig.
f_1	15	33104	0.000	-1.16	1.00	1.000	28500	1.00	16752	0.000	1.00	1.000
	30	73572	0.000	-1.34	1.00	1.000	54797	1.00	19323	0.000	1.00	1.000
	60	165341	0.000	-1.54	1.00	1.000	107657	1.00	20001	0.000	1.00	1.000
f_2	15	35750	0.000	-1.16	1.00	1.000	30786	1.00	16753	0.000	1.00	1.000
	30	81342	0.000	-1.35	1.00	1.000	60091	1.00	19942	0.000	1.00	1.000
	60	194560	0.000	-1.55	1.00	1.000	125703	1.00	23078	0.000	1.00	1.000
f_3	10	40816	0.000	-1.26	1.00	1.000	32479	1.00	11530	0.000	1.00	1.000
	20	153919	0.000	-1.48	1.00	1.000	104266	1.00	9946	0.000	1.00	1.000
	40	856337	0.000	-1.77	1.00	1.000	484506	1.00	11247	0.000	1.00	1.000
f_4	15	114493	0.000	-1.08	1.00	0.025	106017	0.95	52927	0.000	1.00	0.025
	30	360106	0.000	1.07	1.00	0.002	386333	0.91	44850	0.000	1.00	0.002
	60	1623293	0.000	-	1.00	0.000	-	0.00	24574	0.000	1.00	0.000
f_5	5	47032	0.000	-3.25	1.00	0.000	14482	0.34	2626	0.000	1.00	0.000
	10	297631	0.000	-	0.93	0.000	-	0.00	3020	0.000	1.00	0.000
	20	721140	0.000	-	0.05	0.025	-	0.00	4012	0.000	1.00	0.000
f_6	15	35114	0.000	-1.17	1.00	1.000	29943	1.00	16777	0.000	1.00	1.000
	30	77655	0.000	-1.36	1.00	1.000	57248	1.00	20107	0.000	1.00	1.000
	60	175135	0.000	-1.54	1.00	1.000	113623	1.00	21967	0.000	1.00	1.000
f_7	15	10345	0.000	-1.30	1.00	1.000	7964	1.00	5486	0.000	1.00	1.000
	30	19142	0.000	-1.60	1.00	1.000	11976	1.00	6083	0.000	1.00	1.000
	60	30203	0.000	-2.12	1.00	0.158	14218	0.98	5991	0.000	1.00	0.158
f_8	15	67437	0.000	-1.17	1.00	1.000	57552	1.00	21967	0.000	1.00	1.000
	30	144154	0.000	-1.36	1.00	1.000	105765	1.00	22135	0.000	1.00	1.000
	60	330425	0.000	-1.57	1.00	0.000	210578	0.86	21141	0.000	1.00	0.000
f_{15}	15	37467	0.000	-1.17	1.00	0.158	31893	0.98	47254	0.000	1.00	0.158
	30	81475	0.000	-1.38	1.00	0.000	58960	0.83	100955	0.000	0.94	0.015
	60	170500	0.000	-1.50	0.33	0.763	113674	0.31	191963	0.000	0.27	0.535
f_{18}	5	15284	0.000	-1.39	0.97	0.000	10988	0.50	7165	0.000	1.00	0.000
	10	186720	0.000	-	0.40	0.000	-	0.00	5385	0.000	1.00	0.000
	20	812243	0.000	-	0.14	0.000	-	0.00	5429	0.000	1.00	0.000
f_{19}	15	76929	0.000	-1.53	1.00	1.000	50418	1.00	7404	0.000	1.00	1.000
	30	356337	0.000	-2.06	1.00	1.000	172694	1.00	5794	0.000	1.00	1.000
	60	-	0.000	-	0.00	0.000	898583	0.12	5714	0.000	1.00	0.000
f_{21}	15	69136	0.000	-1.19	1.00	1.000	58157	1.00	15146	0.000	1.00	1.000
	30	155494	0.000	-1.46	1.00	1.000	106165	1.00	11334	0.000	1.00	1.000
	60	674925	0.000	-1.99	0.56	0.000	339772	0.88	11852	0.000	1.00	0.000
f_{22}	15	110573	0.000	-1.45	1.00	1.000	76476	1.00	19735	0.000	1.00	1.000
	30	-	-	-	0.00	1.000	-	0.00	31873	0.000	1.00	0.000
	60	-	-	-	0.00	1.000	-	0.00	44988	0.000	1.00	0.000
f_{23}	15	8281	0.000	-1.14	1.00	1.000	7266	1.00	4860	0.000	1.00	1.000
	30	19856	0.000	-1.41	1.00	1.000	14079	1.00	6454	0.000	1.00	1.000
	60	79102	0.243	-	0.95	1.000	106222	0.95	7460	0.000	1.00	1.000
f_{24}	15	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	30	-	-	-	0.00	1.000	-	0.00	-	-	0.00	0.000
	60	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
f_{30}	15	17349	0.000	-1.23	1.00	1.000	14145	1.00	8794	0.000	1.00	1.000
	30	41928	0.000	-1.39	1.00	1.000	30141	1.00	11764	0.000	1.00	1.000
	60	148740	0.000	-1.98	1.00	0.000	75122	0.78	14796	0.000	1.00	0.000
f_{31}	15	217411	0.000	-3.73	1.00	0.000	58290	0.82	10767	0.000	1.00	0.000
	30	340427	0.000	-3.31	1.00	0.000	102760	0.79	12944	0.000	1.00	0.000
	60	694462	0.000	-2.36	0.71	0.028	294531	0.84	15819	0.000	1.00	0.000
f_{41}	5	6741	0.000	-1.10	1.00	1.000	6149	1.00	5802	0.002	1.00	1.000
	10	16074	0.000	-1.11	1.00	1.000	14479	1.00	11522	0.000	1.00	1.000
	20	38851	0.000	-1.23	1.00	1.000	31650	1.00	14766	0.000	1.00	1.000
f_{51}	5	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	10	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	20	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
f_{52}	5	8211	0.000	-1.13	1.00	1.000	7261	1.00	6758	0.000	1.00	1.000
	10	21944	0.000	-1.15	1.00	1.000	19145	1.00	14551	0.000	1.00	1.000
	20	92199	0.000	-1.40	1.00	1.000	65713	1.00	53560	0.000	1.00	1.000
f_{56}	5	-	-	-	0.00	1.000	-	0.00	4626	0.000	1.00	0.000
	10	-	-	-	0.00	1.000	-	0.00	4547	0.000	1.00	0.000

Table 1: Continue

		DE							IODE			
		NFC			SR		COODE		NFC		SR	
Function	D	Avg	Sig.	AR	Avg	Sig.	NFC	SR	Avg	Sig.	Avg	Sig.
	20	-	-	-	0.00	1.000	-	0.00	8865	0.000	1.00	0.000
Average	D/2			-1.48	0.86			0.79				
	D			-1.55	0.78			0.69				
	2D			-1.71	0.61			0.56				

D: Dimension, DE: Differential evolution, COODE: Opposition-based differential evolution using the current optimum, IODE: Improved version of opposition-based differential evolution, AR: Acceleration rate

Table 2: Comparison of DE, COODE and IODE for different population sizes

		DE							IODE			
		NFC			SR		COODE		NFC		SR	
Function	N_p	Avg	Sig.	AR	Avg	Sig.	NFC	SR	Avg	Sig.	Avg	Sig.
f_1	50	34233	0.000	-1.46	1.00	0.000	23452	0.88	8520	0.000	1.00	0.000
	100	73572	0.000	-1.34	1.00	1.000	54797	1.00	19323	0.000	1.00	1.000
	200	205504	0.000	-1.34	1.00	1.000	132970	1.00	35600	0.000	1.00	1.000
f_2	50	40877	0.000	-1.58	1.00	0.001	25939	0.90	8513	0.000	1.00	0.001
	100	81342	0.000	-1.35	1.00	1.000	60091	1.00	19942	0.000	1.00	1.000
	200	227642	0.000	-1.54	1.00	1.000	147396	1.00	39162	0.000	1.00	1.000
f_3	50	91496	0.000	-1.16	1.00	1.000	78815	1.00	5608	0.000	1.00	1.000
	100	153919	0.000	-1.48	1.00	1.000	104266	1.00	9946	0.000	1.00	1.000
	200	430744	0.000	-2.21	1.00	1.000	195038	1.00	16094	0.000	1.00	1.000
f_4	50	-	-	-	0.00	1.000	-	0.00	10162	0.000	0.82	0.000
	100	360106	0.000	1.07	1.00	0.002	386333	0.91	44850	0.000	1.00	0.002
	200	725570	0.000	-1.33	1.00	1.000	546574	1.00	51230	0.000	1.00	1.000
f_5	50	73013	0.000	-	0.46	0.000	-	0.00	1487	0.000	1.00	0.000
	100	297631	0.000	-	0.93	0.000	-	0.00	3020	0.000	1.00	0.000
	200	889522	0.000	-	0.23	0.000	-	0.00	5850	0.000	1.00	0.000
f_6	50	35105	0.000	-1.44	1.00	0.000	24347	0.88	9335	0.000	1.00	0.000
	100	77655	0.000	-1.36	1.00	1.000	57248	1.00	20107	0.000	1.00	1.000
	200	216462	0.000	-1.53	1.00	1.000	141078	1.00	36510	0.000	1.00	1.000
f_7	50	6703	0.000	-1.42	1.00	0.025	4730	0.95	2604	0.000	1.00	0.025
	100	19142	0.000	-1.60	1.00	1.000	11976	1.00	6083	0.000	1.00	1.000
	200	53418	0.000	-1.92	1.00	1.000	27882	1.00	12442	0.000	1.00	1.000
f_8	50	71963	0.000	-1.59	0.94	0.000	45313	0.64	12351	0.000	1.00	0.000
	100	144154	0.000	-1.36	1.00	1.000	105765	1.00	22135	0.000	1.00	1.000
	200	407074	0.000	-1.57	1.00	1.000	259798	1.00	36722	0.000	1.00	1.000
f_9	50	1657	0.000	-1.10	1.00	1.000	1500	1.00	1500	0.987	1.00	1.000
	100	3227	0.000	-1.10	1.00	1.000	2940	1.00	2955	0.750	1.00	1.000
	200	6146	0.000	-1.09	1.00	1.000	5640	1.00	5558	0.346	1.00	1.000
f_{10}	50	17720	0.018	-2.95	1.00	0.000	6009	0.75	4083	0.000	1.00	0.000
	100	13632	0.000	-1.18	1.00	1.000	11539	1.00	8164	0.000	1.00	1.000
	200	27156	0.000	-1.29	1.00	1.000	20974	1.00	17254	0.000	1.00	1.000
f_{11}	50	2706	0.000	-1.19	1.00	1.000	2274	1.00	1632	0.000	1.00	1.000
	100	5313	0.000	-1.16	1.00	1.000	4568	1.00	3746	0.000	1.00	1.000
	200	10372	0.000	-1.17	1.00	1.000	8894	1.00	8146	0.000	1.00	1.000
f_{12}	50	2083	0.000	-1.10	1.00	1.000	1891	1.00	1870	0.484	1.00	1.000
	100	4127	0.000	-1.09	1.00	1.000	3770	1.00	3703	0.205	1.00	1.000
	200	8036	0.000	-1.09	1.00	1.000	7378	1.00	7434	0.449	1.00	1.000
f_{13}	50	5188	0.000	-1.23	0.83	0.000	4210	0.60	3800	0.000	1.00	0.000
	100	10889	0.000	-1.20	0.92	0.000	9078	0.68	7730	0.000	1.00	0.000
	200	21596	0.000	-1.15	0.97	0.000	18775	0.79	16552	0.000	1.00	0.000
f_{14}	50	2234	0.000	-1.37	1.00	1.000	1633	1.00	1659	0.558	1.00	1.000
	100	4815	0.000	-1.47	1.00	1.000	3284	1.00	3243	0.549	1.00	1.000
	200	9496	0.000	-1.45	1.00	1.000	6568	1.00	6386	0.156	1.00	1.000
f_{15}	50	34195	0.000	-1.37	0.58	0.571	24983	0.54	37352	0.000	0.84	0.000
	100	81475	0.000	-1.38	1.00	0.000	58960	0.83	100955	0.000	0.94	0.015
	200	229360	0.000	-1.58	1.00	0.002	145334	0.91	275308	0.000	0.98	0.030
f_{16}	50	1408	0.001	-1.05	1.00	1.000	1345	1.00	1338	0.786	1.00	1.000
	100	2679	0.000	-1.05	1.00	1.000	2543	1.00	2628	0.046	1.00	1.000
	200	5174	0.000	-1.05	1.00	1.000	4932	1.00	5078	0.027	1.00	1.000
f_{17}	50	-	-	-	0.00	1.000	-	0.00	119100	0.000	0.04	0.045

Table 2: Continue

		DE							IODE			
		NFC			SR		CODE		NFC		SR	
Function	N_p	Avg	Sig.	AR	Avg	Sig.	NFC	SR	Avg	Sig.	Avg	Sig.
	100	-	0.000	-	0.00	0.045	37325	0.04	-	0.000	0.00	0.045
	200	-	0.000	-	0.00	0.002	46022	0.09	-	0.000	0.00	0.002
f_{18}	50	58006	0.000	-	0.17	0.000	-	0.00	2893	0.000	1.00	0.000
	100	186720	0.000	-	0.40	0.000	-	0.00	5385	0.000	1.00	0.000
	200	466711	0.000	-	0.83	0.000	-	0.00	8402	0.004	1.00	0.000
f_{19}	50	416072	0.000	-2.38	0.93	0.449	174493	0.90	2940	0.000	1.00	0.001
	100	356337	0.000	-2.06	1.00	1.000	172694	1.00	5794	0.000	1.00	1.000
	200	-	0.000	-	0.00	0.000	305946	1.00	10082	0.000	1.00	1.000
f_{20}	50	2745	0.000	-1.53	1.00	1.000	1792	1.00	1707	0.131	1.00	1.000
	100	6345	0.000	-1.73	1.00	1.000	3666	1.00	3370	0.003	1.00	1.000
	200	13244	0.000	-1.86	1.00	1.000	7102	1.00	6736	0.020	1.00	1.000
f_{21}	50	59644	0.000	-1.36	1.00	1.000	43863	1.00	8857	0.000	1.00	1.000
	100	155494	0.000	-1.46	1.00	1.000	106165	1.00	11334	0.000	1.00	1.000
	200	448128	0.000	-1.69	1.00	1.000	264826	1.00	15674	0.000	1.00	1.000
f_{22}	50	-	-	-	0.00	1.000	-	0.00	15732	0.000	1.00	0.000
	100	-	-	-	0.00	1.000	-	0.00	31873	0.000	1.00	0.000
	200	946995	0.000	-2.32	1.00	0.000	408543	0.42	50064	0.000	1.00	0.000
f_{23}	50	8519	0.070	-	1.00	1.000	10596	1.00	2891	0.000	1.00	1.000
	100	19856	0.000	-1.41	1.00	1.000	14079	1.00	6454	0.000	1.00	1.000
	200	54580	0.000	-1.67	1.00	1.000	32648	1.00	14966	0.000	1.00	1.000
f_{24}	50	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	100	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	200	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
f_{25}	50	5853	0.295	-	1.00	0.025	5512	0.95	5762	0.445	1.00	0.025
	100	9880	0.000	-1.08	1.00	1.000	9147	1.00	9664	0.126	1.00	1.000
	200	18692	0.000	-1.10	1.00	1.000	17006	1.00	18920	0.001	1.00	1.000
f_{26}	50	4592	0.000	-1.18	1.00	0.000	3886	0.78	3469	0.000	1.00	0.000
	100	9636	0.000	-1.19	1.00	0.002	8105	0.91	6670	0.000	1.00	0.002
	200	18978	0.000	-1.17	1.00	0.025	16173	0.95	14234	0.000	1.00	0.025
f_{27}	50	4448	0.000	-1.16	1.00	0.000	3849	0.84	3426	0.000	1.00	0.000
	100	9025	0.000	-1.14	1.00	0.014	7883	0.94	7033	0.000	1.00	0.014
	200	17610	0.000	-1.11	1.00	1.000	15868	1.00	14644	0.000	1.00	1.000
f_{28}	50	4427	0.000	-1.13	1.00	0.001	3919	0.90	3331	0.000	1.00	0.001
	100	8876	0.000	-1.13	1.00	1.000	7880	1.00	6885	0.000	1.00	1.000
	200	17798	0.000	-1.14	1.00	1.000	15560	1.00	14414	0.000	1.00	1.000
f_{29}	50	4199	0.000	-1.23	0.91	0.000	3424	0.60	2946	0.000	1.00	0.000
	100	8455	0.000	-1.20	1.00	0.000	7051	0.76	5672	0.000	1.00	0.000
	200	16782	0.000	-1.20	1.00	0.008	13944	0.93	12920	0.000	1.00	0.008
f_{30}	50	22797	0.000	-1.55	0.47	0.255	14747	0.39	5510	0.000	1.00	0.000
	100	41928	0.000	-1.39	1.00	1.000	30141	1.00	11764	0.000	1.00	1.000
	200	118116	0.000	-1.66	1.00	1.000	71282	1.00	26098	0.000	1.00	1.000
f_{31}	50	72691	0.000	-1.45	1.00	0.000	50213	0.78	10093	0.000	1.00	0.000
	100	340427	0.000	-3.31	1.00	0.000	102760	0.79	12944	0.000	1.00	0.000
	200	-	0.000	-	0.00	0.002	251479	0.91	16528	0.000	1.00	0.002
f_{32}	50	2580	0.000	-1.37	1.00	1.000	1877	1.00	1363	0.000	1.00	1.000
	100	4916	0.000	-1.36	1.00	1.000	3625	1.00	3167	0.000	1.00	1.000
	200	9676	0.000	-1.37	1.00	1.000	7062	1.00	6200	0.000	1.00	1.000
f_{33}	50	12297	0.000	-2.02	1.00	0.000	6077	0.13	1122	0.000	1.00	0.000
	100	25696	0.000	-1.92	1.00	1.000	13398	1.00	2499		1.00	1.000
	200	51934	0.000	-1.92	1.00	0.000	26982	0.87	4884	0.000	1.00	0.000
f_{34}	50	13664	0.000	-2.28	0.89	0.000	5996	0.12	1090	0.000	1.00	0.000
	100	30552	0.000	-2.40	1.00	0.000	12744	0.41	2495	0.000	1.00	0.000
	200	64826	0.000	-2.46	1.00	0.000	26349	0.63	5222	0.000	1.00	0.000
f_{35}	50	1572	0.000	-1.10	1.00	1.000	1432	1.00	1468	0.101	1.00	1.000
	100	3005	0.000	-1.08	1.00	1.000	2792	1.00	2859	0.148	1.00	1.000
	200	5890	0.000	-1.08	1.00	1.000	5434	1.00	5676	0.000	1.00	1.000
f_{36}	50	3304	0.000	-1.89	1.00	1.000	1748	1.00	1829	0.140	1.00	1.000
	100	7556	0.000	-2.02	1.00	1.000	3742	1.00	3349	0.001	1.00	1.000
	200	16050	0.000	-2.13	1.00	1.000	7544	1.00	7100	0.013	1.00	1.000
f_{37}	50	2105	0.000	-1.07	1.00	1.000	1959	1.00	1924	0.373	1.00	1.000
	100	4120	0.000	-1.06	1.00	1.000	3898	1.00	3825	0.235	1.00	1.000
	200	8046	0.000	-1.05	1.00	1.000	7634	1.00	7666	0.698	1.00	1.000
f_{38}	50	2101	0.000	-1.09	1.00	1.000	1933	1.00	1875	0.204	1.00	1.000

Table 2: Continue

		DE							IODE			
		NFC			SR		COODE		NFC		SR	
Function	N_p	Avg	Sig.	AR	Avg	Sig.	NFC	SR	Avg	Sig.	Avg	Sig.
	100	4117	0.000	-1.06	1.00	1.000	3866	1.00	3777	0.195	1.00	1.000
	200	8004	0.000	-1.06	1.00	1.000	7580	1.00	7724	0.024	1.00	1.000
f_{39}	50	1517	0.000	-1.07	1.00	1.000	1416	1.00	1409	0.772	1.00	1.000
	100	2874	0.000	-1.06	1.00	1.000	2715	1.00	2759	0.394	1.00	1.000
	200	5600	0.000	-1.06	1.00	1.000	5270	1.00	5442	0.015	1.00	1.000
f_{40}	50	3354	0.000	-1.37	1.00	1.000	2453	1.00	2377	0.218	1.00	1.000
	100	7184	0.000	-1.46	1.00	1.000	4920	1.00	4743	0.075	1.00	1.000
	200	14686	0.000	-1.51	1.00	1.000	9746	1.00	9298	0.019	1.00	1.000
f_{41}	50	7241	0.000	-1.14	1.00	1.000	6343	1.00	4944	0.000	1.00	1.000
	100	16074	0.000	-1.11	1.00	1.000	14479	1.00	11522	0.000	1.00	1.000
	200	33706	0.000	-1.05	1.00	1.000	32046	1.00	23928	0.000	1.00	1.000
f_{42}	50	1850	0.000	-1.09	1.00	1.000	1700	1.00	1644	0.124	1.00	1.000
	100	3579	0.000	-1.09	1.00	1.000	3284	1.00	3333	0.303	1.00	1.000
	200	6944	0.000	-1.07	1.00	1.000	6472	1.00	6422	0.511	1.00	1.000
f_{43}	50	2782	0.000	-1.12	1.00	1.000	2474	1.00	2350	0.055	1.00	1.000
	100	5468	0.000	-1.14	1.00	1.000	4811	1.00	4756	0.512	1.00	1.000
	200	10622	0.000	-1.13	1.00	1.000	9380	1.00	9344	0.793	1.00	1.000
f_{44}	50	2800	0.000	-1.10	1.00	1.000	2540	1.00	2440	0.071	1.00	1.000
	100	5520	0.000	-1.07	1.00	1.000	5181	1.00	5089	0.241	1.00	1.000
	200	10854	0.000	-1.07	1.00	1.000	10100	1.00	9974	0.437	1.00	1.000
f_{45}	50	1354	0.000	-1.12	1.00	1.000	1214	1.00	1252	0.075	1.00	1.000
	100	2549	0.000	-1.06	1.00	1.000	2416	1.00	2434	0.597	1.00	1.000
	200	4888	0.000	-1.05	1.00	1.000	4642	1.00	4696	0.383	1.00	1.000
f_{46}	50	2457	0.000	-1.10	1.00	1.000	2224	1.00	2059	0.002	1.00	1.000
	100	4826	0.000	-1.08	1.00	1.000	4470	1.00	4302	0.054	1.00	1.000
	200	9494	0.000	-1.09	1.00	1.000	8740	1.00	8608	0.351	1.00	1.000
f_{47}	50	1375	0.000	-1.08	1.00	1.000	1269	1.00	1293	0.278	1.00	1.000
	100	2610	0.000	-1.06	1.00	1.000	2453	1.00	2535	0.018	1.00	1.000
	200	5018	0.000	-1.06	1.00	1.000	4714	1.00	4870	0.017	1.00	1.000
f_{48}	50	1340	0.000	-1.15	1.00	1.000	1167	1.00	1610	0.000	1.00	1.000
	100	2475	0.000	-1.12	1.00	1.000	2202	1.00	3059	0.000	1.00	1.000
	200	4656	0.000	-1.13	1.00	1.000	4124	1.00	5798	0.000	1.00	1.000
f_{49}	50	2253	0.000	-1.22	0.71	0.759	1847	0.69	1752	0.169	0.87	0.002
	100	4418	0.000	-1.17	0.94	0.092	3783	0.87	3719	0.447	1.00	0.000
	200	8590	0.000	-1.16	1.00	0.045	7417	0.96	7133	0.057	1.00	0.045
f_{50}	50	602744	0.001	-3.51	0.16	0.083	171594	0.08	97201	0.258	1.00	0.000
	100	242434	0.000	-1.99	1.00	0.000	121722	0.86	18131	0.000	1.00	0.000
	200	460328	0.000	-2.69	1.00	0.000	171366	0.88	11756	0.000	1.00	0.000
f_{51}	50	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	100	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	200	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
f_{52}	50	9914	0.000	-1.18	1.00	1.000	8378	1.00	6745	0.000	1.00	1.000
	100	21944	0.000	-1.15	1.00	1.000	19145	1.00	14551	0.000	1.00	1.000
	200	46022	0.000	-1.10	1.00	1.000	41770	1.00	29920	0.000	1.00	1.000
f_{53}	50	3490	0.000	-1.66	1.00	0.008	2104	0.93	1246	0.000	1.00	0.008
	100	6834	0.000	-1.63	1.00	1.000	4199	1.00	2771	0.000	1.00	1.000
	200	13214	0.000	-1.60	1.00	1.000	8236	1.00	5626	0.000	1.00	1.000
f_{54}	50	3671	0.000	-1.10	1.00	1.000	3332	1.00	2882	0.000	1.00	1.000
	100	7369	0.000	-1.11	1.00	1.000	6636	1.00	6049	0.000	1.00	1.000
	200	14290	0.000	-1.08	1.00	1.000	13172	1.00	11814	0.000	1.00	1.000
f_{55}	50	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	100	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
	200	-	-	-	0.00	1.000	-	0.00	-	-	0.00	1.000
f_{56}	50	-	-	-	0.00	1.000	-	0.00	2802	0.000	1.00	0.000
	100	-	-	-	0.00	1.000	-	0.00	4547	0.000	1.00	0.000
	200	-	-	-	0.00	1.000	-	0.00	13106	0.000	1.00	0.000
f_{57}	50	7774	0.000	-1.16	1.00	1.000	6704	1.00	4032	0.000	1.00	1.000
	100	15458	0.000	-1.15	1.00	1.000	13483	1.00	10055	0.000	1.00	1.000
	200	30754	0.000	-1.15	1.00	1.000	26858	1.00	21960	0.000	1.00	1.000
f_{58}	50	42750	0.005	-7.52	1.00	0.000	5686	0.77	3799	0.000	1.00	0.000

Table 2: Continue

		DE							IODE			
		NFC			SR		COODE		NFC		SR	
Function	N_p	Avg	Sig.	AR	Avg	Sig.	NFC	SR	Avg	Sig.	Avg	Sig.
	100	13592	0.000	-1.23	1.00	1.000	11035	1.00	08139	0.000	1.00	1.000
	200	27084	0.000	-1.23	1.00	1.000	21016	1.00	17444	0.000	1.00	1.000
Average	$N_p/2$			-1.41	0.81			0.72				
	N_p			-1.36	0.88			0.83				
	$2N_p$			-1.40	0.86			0.85				

N_p: Dimension, DE: Differential evolution, COODE: Opposition-based differential evolution using the current optimum, IODE: Improved version of opposition-based differential evolution, AR: Acceleration rate

D = 20, f_{18} for D = 20 but DE solves it in 100, 5 and 14% of the trials, respectively. On the other hand, DE cannot solve f_{19} for D = 60 but COODE solves it in 12% of the trials.

In the bottom of Table 1, the average SR and the average AR for functions with D/2, D and 2D, are presented. For functions with dimension of D/2, the overall SR for DE is 7% higher than COODE's (0.86 vs., 0.79) and the overall AR is 1.48. For functions with dimension of 2D, the overall AR is 1.71 and the SR for DE and COODE are 0.61 and 0.56, respectively.

Decreasing the overall SR for DE and COODE was predictable because by doubling the problem dimension, algorithms are sometimes unable to solve the problem before reaching the maximum NFC (which is a fixed number for all experiments). However, as seen from Table 1, COODE performs better for high-dimensional problems. The higher average AR belongs to the functions with dimension 2D. According to Table 1, for all of functions with different dimensions, IODE performs best among all tested algorithms.

Experiment series 2: Effect of population size: In order to investigate the effect of the population size, the same experiments are repeated for $N_p/2$ and $2N_p$ and the overall results for different population size are summarized in Table 2. The better results of the NFC and the SR for different population size are highlighted in boldface.

According to Table 2, for the majority of functions (excepting function f_4 (N_p = 100), f_5 (N_p = 50, 100 and 200) and f_{18} (N_p = 50, 100 and 200)), COODE performs better than DE. The average AR for different population size is -1.41, -1.36 and -1.40, respectively. There is no difference for different cases.

For $N_p/2$, DE fails to solve 7 functions (f_4, f_{17}, f_{22}, f_{24}, f_{51}, f_{55} and f_{56}) while COODE fails on 10 (f_4, f_5, f_{17}, f_{18}, f_{22}, f_{24}, f_{50}, f_{51}, f_{55} and f_{56}). However, the average SR for DE and COODE is 0.81 and 0.72, respectively. For $2N_p$, this number for DE and COODE is 7, 7, 0.86 and 0.85, respectively. Obviously, two algorithms are more likely to succeed when we select the larger population size. According to the results of this section, COODE performs better than DE in the term of success rate for large population size.

Then, the performance of COODE and IODE are compared carefully for all cases. It is inspiring to note that there is no statistically difference in NFC for 44 functions which means that the results obtained by COODE nearly reach the calculated the theoretical optimal value. However, for 14 functions (f_{15} (N_p = 50, 100 and 200), f_{16} (N_p = 100 and 200), f_{25} (N_p = 200), f_{35} (N_p = 200), f_{38} (N_p = 200), f_{39} (N_p = 200), f_{47} (N_p = 100 and 200) and f_{48} (N_p = 50, 100 and 200)), COODE performances even better than IODE. Obviously, it is contradictory with the design principles of IODE. The reason is unknown until now but is probably related to some features of benchmark functions.

According to the results of COODE (the ninth and tenth rows in Table 2), the average NFC increases as the population size increases while the average success rate in this study also increases from 72-85%. Therefore, it is better to adopt a compromising position on population size, reducing the number of function calls and enhancing the success rate of COODE algorithm, to meet the demands of engineering application.

CONCLUSION

In this study, DE, COODE and IODE were compared for different problem dimensions (D/2, D and 2D) and different population sizes ($N_p/2$, N_p and $2N_p$). COODE performs better than DE for larger population size which is usually required for more complex and high-dimensional problems.

Utilizing opposite points to accelerate an optimization method is a new idea. The future work should be done on applying the COOBL scheme to accelerate other methods, such as genetic algorithm, evolutionary programming, artificial neural network and particle swarm optimization. Another interesting work is to solve large scale optimization problem, optimization of noisy problem, multi-objective optimization, traveling salesman problem, image processing and understanding and other engineering problems.

ACKNOWLEDGMENT

This study was supported by the National Natural Science Foundation of China (No. 61100009).

REFERENCES

Dong, N. and Y.P. Wang, 2009. Multiobjective differential evolution based on opposite operation. Proceedings of the International Conference on Computational Intelligence and Security, December 11-14, 2009, Beijing, China, pp: 123-127.

Han, L. and X.S. He, 2007. A novel opposition-based particle swarm optimization for noisy problems. Proceedings of the 3rd International Conference on Natural Computation, August 24-27, 2007, Haikou, China, pp: 624-629.

Kanzawa, Y., Y. Endo and S. Miyamoto, 2007. Fuzzy c-means algorithms for data with tolerance based on opposite criterions. IEICE Trans. Fundamen. Electron. Commun. Comput. Sci., E90: 2194-2202.

Malisia, A.R. and H.R. Tizhoosh, 2007. Applying opposition-based ideas to the ant colony system. Proceedings of the IEEE Swarm Intelligence Symposium, April 1-5, 2007, Honolulu, USA., pp: 182-189.

Peng, L., Y.Z. Wang and G.M. Dai, 2008. A novel opposition-based multi-objective differential evolution algorithm for multi-objective optimization. Adv. Comput. Intelli., 5370: 162-170.

Rahnamayan, S., H.R. Tizhoosh and M.M.A. Salama, 2006. Opposition-based differential evolution algorithms. Proceeding of the IEEE Congress on Evolutionary Computation, July 16-21, 2006, Vancourver, Canada, pp: 2010-2017.

Rahnamayan, S. and G.G. Wang, 2008. Solving large scale optimization problems by Opposition-based Differential Evolution (ODE). WSEAS Trans. Comput., 7: 1792-1804.

Rahnamayan, S., H.R. Tizhoosh and M.M.A. Salama, 2008a. Opposition versus randomness in soft computing techniques. Applied Soft Comput., 8: 906-918.

Rahnamayan, S., H.R. Tizhoosh and M.M.A. Salama, 2008b. Opposition-based differential evolution. IEEE Trans. Evolut. Comput., 12: 64-79.

Rashid, M. and A.R. Baig, 2010. Improved opposition-based PSO for feedforward neural network training. Proceeding of the International Conference on Information Science and Applications, April 21-23, 2010, Seoul, Korea, pp: 1-6.

Subudhi, B. and D. Jena, 2011. A differential evolution based neural network approach to nonlinear system identification. Applied Soft Comput., 11: 861-871.

Tizhoosh, H.R., 2005. Opposition-based learning: A new scheme for machine intelligence. Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, Volume 1, November 28-30, 2005, Vienna, Austria, pp: 695-701.

Tizhoosh, H.R., M. Ventresca and S. Rahnamayan, 2008. Opposition-Based Computing. In: Studies in Computational Intelligence: Oppositional Concepts in Computational Intelligence, Tizhoosh, H.R. and M. Ventresca (Eds.). Springer, USA., pp: 11-28.

Wang, H., Z.J. Wu and S. Rahnamayan, 2011. Enhanced opposition-based differential evolution for solving high-dimensional continuous optimization problems. Soft Comput.- A Fusion Foundat. Methodol. Appli., 15: 2127-2140.

Xu, Q., L. Wang, B. He and N. Wang, 2011a. Modified opposition-based differential evolution for function optimization. J. Computat. Inf. Syst., 7: 1582-1591.

Xu, Q.Z., L. Wang, B.M. He and N. Wang, 2011b. Opposition-based differential evolution using the current optimum for function optimization. J. Applied Sci., 29: 308-315.