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

The Monarchy Driven Optimization Technique

Ritambhar Burman[1], Swagatam Das[2], Zeeshanul Haque[1], Athanasios V. Vasilakos, and Soumyadeep Chakrabarti[1]

[1]Dept. of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata 700 032, India.
[2]Electronics and Communication Sciences Unit, Indian Statistical Institute, Kolkata 700 108, India.
E-mail: ritambharkcdburman@gmail.com, mrzeeshanulhaque@gmail.com, soumyadeepchakrabarti@gmail.com, swagatam.das@isical.ac.in

Abstract — **We present a novel human society inspired algorithm for solving single-objective bound constrained optimization problems. The proposed Monarchy Driven Optimization (MDO) algorithm is a population-based iterative global optimization technique for multi-dimensional and multi-modal problems. At its core, this technique introduces a monarchial society where the outlook of its population is fashioned by the thoughts of individuals and the monarch. A detailed study including the tuning of MDO parameters is presented along with the theory. It is applied to standard benchmark functions comprising uni-modal and multi-modal as well as rotated functions. The results section suggests that, in most instances, MDO outperforms other well-known techniques such as Particle Swarm Optimization (PSO), Differential Evolution (DE), Gravitational Search Algorithm (GSA), Comprehensive Learning Particle Swarm Optimization (CLPSO) and Artificial Bee Colony (ABC) optimization in terms of final convergence value and mean convergence value, thus proves to be a robust optimization technique.**

Keywords—monarch, society, thought, outlook, peak outlook, peak thought.

I. INTRODUCTION

Various optimization techniques have been developed in recent past by harnessing nature's activities. Techniques inspired from the Darwinian evolution process like Genetic Algorithm (GA) [1] have proven to be very efficient global optimization methods. Further, Particle Swarm Optimization (PSO) [2-3], Ant Colony Optimization (ACO) [4-6], Differential Evolution(DE) [7-9], Artificial Bee Colony Optimization [10], Comprehensive Learning Particle Swarm Optimization (CLPSO) [11], Honey Bee Optimization (HBO) [12], Imperialist Competitive Algorithm (ICA) [13], Gravitational Search Algorithm (GSA) [14] and recently proposed Wind Driven Optimization (WDO) [15] have showed various ways how nature and its elements can be imitated through simulation to develop new schemes for solving mathematically intractable numerical as well as combinatorial optimization problems. However, each method has its own pros and cons, as suggested by the No Free Lunch theorem [16], which suggests that no particular optimization algorithm can stand out as the best for solving all types of optimization problems.

Generally swarm intelligence has been exploited based on collective behaviours of simple insects or animals like ants, bees, and bacteria. Their collaboration and cooperation can form an intelligent environment. However, it should be more promising to exploit human behaviour as humans are accepted as the most intelligent species on earth. Human psychology can be studied for developing optimization techniques to tackle difficult problems in science and engineering domains. This has led to exploitation of different aspects of human behaviour and thought processes first realized by ICA [13] and then the Brain Storm Optimization (BSO) [17] technique.

Following this line of research we propose a novel population-based iterative heuristic global optimization technique for multi-dimensional and multimodal function optimization problems. Building on the successful record of the existing nature-inspired optimization algorithms, this paper introduces and utilizes an entirely new optimization method which is named as the Monarchy Driven Optimization (MDO) technique, keenly studying how in earlier times societies used to develop under the guidance of their monarchs who effectively ruled their kingdoms. In ICA [12], the countries of the world (analogous to the societies in MDO) are segregated into several empires which have their own imperialists (analogous to monarchs in MDO). It is the duty of the imperialist to lead the colonies under their rule to a better state. The colonies physically move towards their individual imperialists and improve their empires if they find better optima. The weaker colonies collapse and join the stronger colonies. An empire becomes non-existent if there are no colonies under it. Unlike ICA, MDO has a single monarch who leads the entire society by his thought process which is manifested by his outlook. The citizens' thoughts are also given importance as the monarch considers them and updates his own thought and outlook accordingly at the end of each iteration.

MDO is different from PSO due to the fact that PSO uses the personal best position and global best position of the particles in the velocity update equation. However for MDO, thought process of citizens, the analogous term of velocity, is updated by the best thought process of the citizen, which was conducive for him in his past experience, and the thought process of the monarch which is thought to be the best amongst the citizens. The best thought process can be determined by the best outlook of the citizen, the analogous term of position in PSO, by the simple fact that the thought process at the time of best outlook of a particular citizen is thought to be the best. Also, PSO uses the relative position in velocity update equation like the position relative to the personal best position and the position relative to the global best position. However MDO does not use relative thought process but the actual best thought process and the thought process of the monarch in the thought update equation of each citizen.

The rest of the paper is organized in the following order. Section II introduces Monarchy Driven Optimization Technique outlining its inspiration and basic steps. Section III shows experimental comparisons with other well known optimization variants like PSO, DE, GSA, ABC and CLPSO and analyses them. Section IV tunes the parameters of the proposed technique to show its robustness. Lastly it is concluded in section V.

II. MONARCHY DRIVEN OPTIMIZATION TECHNIQUE(MDO)

A. Inspiration

The inspiration for Monarchy Driven Optimization technique comes from the human social behavior significant during pre-democratic era under the rule of monarchs. Monarchy is a form of government in which sovereignty is actually or nominally embodied in a single individual. Here one person takes all administrative decisions. One form of monarchy is absolute monarchy which is described by a monarch with no or few legal restraints in state and political matters. The monarch rules with absolute power over the state and government such as the right to rule by decree and promulgate laws.

A society is a resultant of its population. It prospers only when its multitudes prosper individually. For this, common people form an integral part of the society and their thoughts and mentality cannot be simply disregarded. It is basically the people's attitude or outlook towards the society which leads to the development of society and if properly shaped can lead to an efficient society.

A society is a conglomeration of different citizens and each person is further characterized by his multifaceted outlook. Diversity in people's nature is an essential factor which contributes to society fitness. A diversified society is resilient to challenges. Even if a person fails to contribute effectively to the betterment of society, he can be motivated or influenced by other people and their monarch. This diversified nature of society is initially helps us to harness the great and unexpected intelligence of different people to obtain a situation where the society should lead to reach its true pinnacle. However for a society to ultimately prosper there should be harmony among the various outlooks of all its citizens. Everyone should have the same ideals which result in the most beneficial contribution to the society.

Studying all these features of the monarchical society, this paper attempts for an optimization technique called Monarchy Driven Optimization Technique.

B. Basic Steps

1) Initialisation

A society comprising of Ncitizens having different background and thought-process serves well for the society initially as it introduces the exploration factor. Each person is attributed with different outlook$o$in different aspects of society. This plethora of outlook of a person and the approach towards society in a different way not only adds diversity to the society but also paves a way for the society to choose which of these characteristics should be properly optimized for the society to flourish. These characteristics of a single citizen are stated as dimension $d$of the outlook.

However an important realization here is that for forming a prospering and well blended society, initially an individual cannot have a drastically different outlook or even a very less diversified outlook. Both these extreme conditions are not very helpful for the society. A citizen having a moderate outlook fits well in the society and sure lead to optimization of the society fitness. A misfit in the society tends to drive the society along an erroneous way. Hence each citizen can have their initial outlook in certain bounds. A maximum outlook and minimum outlook is thus defined and each citizen's outlook is randomly chosen within this range.

$$o \leftarrow o_{min} + rand(N,d) \times (o_{max} - o_{min}) \quad (1)$$

where o_{max}ando_{min} are chosen asa suitable range for the proper functioning of society and varies from problem to problem. It spread uniformly over this range, to give the maximum diversification power to the society. This most extreme outlook allowed is an input according to the objective function the optimisation of which is the ultimate goal of the society. A complex society requires more coherent nature in the citizens' mentality to optimise in a better way. This effectively reflects in our method where depending on the complexity of the objective function (or the society welfare) the optimisation can only succeed if a proper range of outlook is chosen.

There is also a restriction on the initial thought process $t$to allow the society to reach a state of harmony quickly, without much anarchy. These are set as half of the respective mentality limitations.

$$t_{max} \leftarrow o_{max}/2 \quad (2)$$

$$t_{min} \leftarrow o_{min}/2 \quad (3)$$

Every citizen is diverse and independent and thus always draws his influence from his own peak thought (hereby also referred to as t_{peak} corresponding to the thought of a citizen when his outlook has been the most favourable for the society referred to as o_{peak}.

$$t^i_{peak} \leftarrow t^i \quad if\ o^i = o^i_{peak} \quad (4)$$

where $i \in \{1,2,3,\dots,N\}$

2) Monarch

A society simply created by different citizens or people cannot flourish effectively without having someone to guide them. Different people with different backgrounds cannot mix together well or more importantly fails to form a harmonized society if they are left like that. Hence a monarch is very much needed. A monarch is the head of states and characterized as a peaceful and just person who takes decision for the welfare of his kingdom (or society under his rule).

In this randomly generated society, the monarch is chosen as the best person among the citizens, who has the best thought and thus the best outlook, and thus best knows what a society needs to flourish. The algorithm for selection of the monarch or the monarch's outlook (here denoted by $o_{monarch}$ only) is simply to evaluate the outlook of all the citizens and selecting the most fit citizen with the best outlook since he is most suitable for becoming the monarch. It is the contribution of these citizens that we are trying to optimize and as the gap of the present society from the optimized society is minimized we say that the fitness value of the objective function has improved. Monarch takes on the best outlook (defined as $o_{monarch}$) as well as best thought (defined as $t_{monarch}$)amongst all the citizens at the end of each iteration.

$$o_{monarch} \leftarrow o^i \quad t_{monarch} \leftarrow t^i \quad , if\ f(o_{monarch}) < f(o^i) \tag{5}$$

3) Alteration of citizen's outlook

A society once formed needs to think of ways through which they can prosper and hence the initially randomized outlook of people needs to be refashioned in a more channelized way so that harmony and success can be brought in the society. This refashioning of citizen's outlook depends on a number of factors which can be enumerated as:

a) Past Thought Process

The citizens' present thought process or the way they think and view things. Without such provision a person's intelligence and more importantly his contribution cannot be judged. So the present thought process is one important factor while considering the refashioning of his outlook.

b) Peak Thought

The peak thought during a citizen's lifetime. Each person flourishes when they can learn from their mistake or even judge the present scenario on the basis of some cornerstone that they have kept in their life. It is a ubiquitous phenomenon that an experienced person knows best what is best for himself and also for the people around him. Thus his experience is an important parameter which properly moulds the society. Each person is attributed by his past experiences, his best experiences and his worst experiences. Although all experiences of the past helps a citizen in some way or another, it is the best past experience represented by the t_{peak} of an individual that needs to be recalled in situations when we are faced with challenges. If we have to optimize a society's fitness it is natural to take into account only the best past peak thought of people.

c) Guidance of the monarch

The rules from their monarch. A society has faith in the monarch that he would lead the society to a better state and thus emulate him. With this hope a citizen updates his outlook and fashions his thought process according to the monarch's guidance. It is thus his duty to think about the people's welfare and also change their outlook in such ways that can draw people closer to him and his ideals which are aimed for betterment to society. Without a monarch the society can fail to harmonize and work together and can eventually turn into chaotic baffled masses. It is thus the duty of the monarch to look into such situation and try to properly change people outlook so that each of them can contribute to the society to their best strength.

It is noteworthy that a society can achieve its optimum best fitness value only when each citizen can contribute to the society and can think along with other people for the betterment of the society and the monarch effectively brings all these people together and can make them favour them. Thus the reshaping of outlook through addition of some thoughts to their present outlook is very necessary.

$$t^{i,d} \leftarrow c_1 * rand * t^{i,d} + c_2 * rand * t^{i,d}_{peak} + c_3 * rand * t^{d}_{monarch} \tag{6}$$

$$o^{i,d} \leftarrow rand * o^{i,d} + t^{i,d} \tag{7}$$

where $t^{i,d}$ represents the d^{th} dimension of the thought process of the i^{th} individual. The thought process of each citizen is effectively added to their individual outlook of the previous iteration to give way for the new outlook of the citizens. The use of rand in (7) conforms to the fact that the factors by which an individual's outlook is influenced by his previous outlook vary from individual to individual and from iteration to iteration.

FIRST TERM OF (6):

It signifies the individuality of the citizen and the coefficient c_1 determines the extent to which a person will follow his own beliefs.Thus c_1 is called willpower coefficient and it is necessary that a person has sufficient amount of it to ensure that he can follow a certain path long enough to ensure the society's gain when the path is right. His own belief implies his thoughts at the previous moment which is analogous to the iterations of various optimization techniques.

SECOND TERM OF (6):

It weighs the past experience of the person. His t_{peak} is something which he has developed over a period of time, and it tells him which thoughts were best for him. To increase the stochastic nature of the simulation, a random number is there in the expression along withc_2, the past experience coefficient.

THIRD TERM OF (6):

The third term corresponds to the driving powers of the monarch, who commands the person to change his thought process. The effect of his leading powers can be measured with the help of c_3 which is hereby called the commanding coefficient.

4) Updating the monarch and mentality of the citizens

At the end of these processes, the peak thought of the individuals are updated by updating their t_{peak} with the new thought if the present outlook is the best outlook they have had in their lifetime. The outlooks$o^{i,d}$ of the citizens are taken as inputs to the benchmark functions which are analogous to position of particles in conventional optimization techniques like PSO and DE. This returns a value indicating the betterment of the society which needs to be minimized for minimization problem and maximized for maximization problems. For minimization problems, less is the cost function, better developed is the society. Here, the fitness value obtained from their here-up-to best outlook, and these o_{peak} are then changed to the corresponding outlooks. The t_{peak} among all these citizens are also updated as mentioned previously. The monarch made conversant with both the o_{peak} and t_{peak} that is the monarch assumes the best outlook and thought corresponding to that outlook, in the society.

$$o^{i}_{peak} \leftarrow o^{i} \quad t^{i}_{peak} \leftarrow t^{i} \quad , if\ f(o^{i}_{peak}) < f(o^{i}) \tag{8}$$

5) Counselling of the monarch

A society cannot develop well if its monarch whom the people follow fails to update himself from time to time. Thus it is required that at the end of regular time periods the monarch sits down with all his citizens and imbibes the best outlook possible to run the society in the best possible manner.

Thus even though the monarch remains the same, he has to keep updating himself with the changing requirements of the society.

$$\begin{matrix} o_{monarch} \leftarrow o_a \\ t_{monarch} \leftarrow t_a \end{matrix} \quad if\ f(o_a) = min(f(o^i)) \forall\ i \tag{9}$$

III. Experimental Verifications and Comparisons

C. Benchmark Functions Tested

Seventeen benchmark functions listed in table II are taken for experimental tests. Table II gives the search range of the defined functions along with the global optimum value, which is the minimum for minimization problems. These benchmark functions are widely adopted in global optimization techniques. The fifteen benchmark functions used are classified into three categories, the first group $f_1 - f_8$containing eight unimodal functions. The second group contains four multimodal functions $f_9 - f_{12}$which are more complexhaving higher dimensionality and multiple local optima. The last group $f_{13} - f_{17}$ contains 5 rotated functions, the last two of which are CEC 2013 functions [18].

D. Experimental Set-up for MDO and other algorithms

The experimental results tabulated in table III performed on the benchmark functions of table II, are done with parameter settings indicated in table I. The parameters involved in MDO technique are c_1, c_2 and c_3 indicated in (6). To test whether the technique is parameter insensitive, each of the three parameters are varied, and the mean error along with standard deviation is computed in table IV and V for both 30 dimension and 50 dimension. The simulation environment is Matlab 2013a in a workstation with Intel Core i5 and 2.50 GHz Processor.

TABLE I: Parameter Setup

Algorithm	Parameter	Setting
MDO	c_1	0.2
	c_2	0.2
	c_3	0.2
PSO	w	0.8
	c_1	1.4
	c_2	1.4
DE	F	0.7
	Cr	0.9
GSA	α	2
	G_0	100
	$Selection\ rate$	0.5
	$Mutation$	0.3
CLPSO	w_0	0.9
	w_1	0.4
	c_1	1.4955
	c_2	1.4955
ABC	Colony size	10
	Employed	5
Dimension		30/50
Citizen/ Particle		20
Ending criteria		20000 Function Evaluation

E. Experimental Results

MDO technique and other well known variants of global optimizers like PSO, DE, GSA, ABC and CLPSO are tested

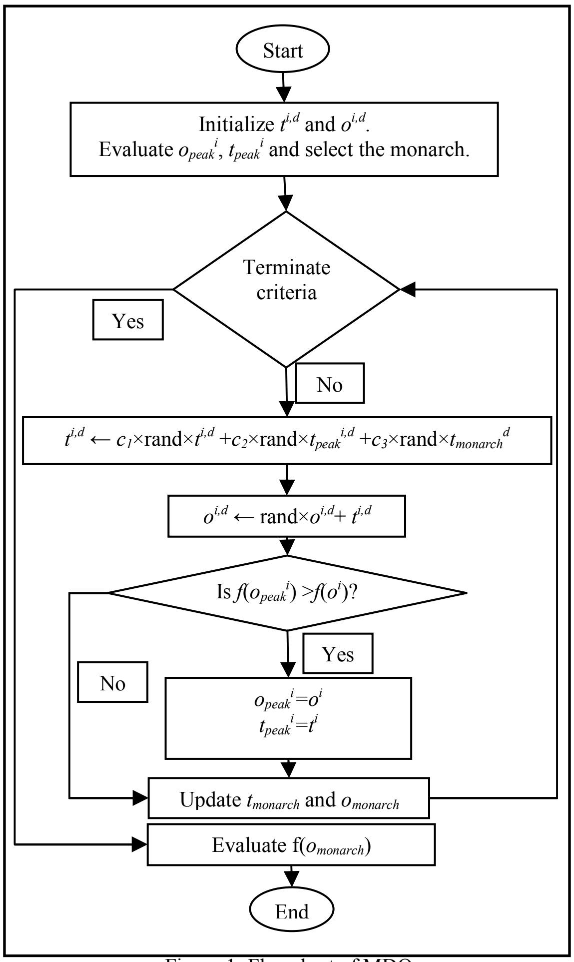

Figure 1: Flowchart of MDO

on the seventeen benchmark functions $f_1 - f_{17}$ of table II. Table III gives the mean error, standard deviation and least error value over a period of 20 runs to remove statistical dependence. Rank is awarded according to the mean error value obtained by the techniques. Best error value means the run securing the least error among the 20 runs for a particular technique. The least among the best error values and least mean error value obtained for the functions are bolded corresponding to the one that obtains it. Error value is defined as $|f(x_{best}) - f(x_{optimum})|$ where $x_{optimum}$ is the position of the particle in MDO located at the global minima for the particular function. The functions are tested over 30 and 50 dimensions, 20 citizens or particles for the counterpart techniques and 20000 FEs as terminating criteria.

F. Comparison

1) Uni-modal functions

Table III depicts that the proposed technique is more efficient than other well known optimization techniques like DE, PSO, GSA and CLPSO for all the eight unimodal benchmark functions $f_1 - f_8$ of table II. Convergence plots of fig 2(a), 2(d) and 2(g) show that the MDO technique descends to the global optima steeply while all the other techniques fail to optimize them decently, except f_8 where the mean and least error value obtained by MDO and GSA are same, whereas DE, CLPSO and PSO obtain the least error value as 0. f_5 and f_7 are the only two functions where MDO technique stagnates before reaching the global minima. But in spite of that, it manages to secure the 1st rank when compared to other optimization techniques. Thus MDO secures the first position for all the eight optimization procedures of this type.

2) Multi-modal functions

Table III depicts that for multi-modal functions $f_9 - f_{12}$, MDO technique is the most suited as the mean error as well as the mean least error found by the different procedures is least for MDO, while standard deviation is 0 for 20 runs. Fig 2(b), 2(e) and 2(h) show that the numerous local minima fail to influence the MDO technique while DE, GSA, PSO and CLPSO get stagnated in the local minima. The MDO technique is capable of finding the global minima for all the test functions except Ackley function f_{10}, where MDO shows mean error 8.88e-16. However it still works better than other algorithms and secures the first rank for f_{10}.

3) Rotated and shifted functions

Functions $f_{13} - f_{17}$ of table III shows that the MDO technique manages to find the global minima of the functions successfully, shown in fig. 2(c), 2(f) and 2(i), in spite of its complicated nature of the functions because of numerous local minima and rotated nature and maintains 100% success rate because of zero standard deviations over a period of 20 runs for $f_{13} - f_{15}$. For the last two test functions $f_{16} - f_{17}$, it is found that it is successful to find the minimum of the shifted function f_{16} and holds the first position, but it secures 2nd and 3rd position respectively for 50 dimensions and 30 dimensional f_{17} function when it comes to find the minima of the function.

TABLE II: Details of the benchmark function

Test function	Minimum	Search Range	Name
$f_1(x) = \sum_{i=1}^{n} x_i^2$	0	[-100,100]	Sphere
$f_2(x) = \sum_{i=1}^{n} \lvert x_i\rvert + \prod_{i=1}^{n} \lvert x_i\rvert$	0	[-10,10]	Schwefel's P2.22
$f_3(x) = x_1^2 + \sum_{i=2}^{n} 10^6 (x_i^2)$	0	[-100,100]	Bent Cigar
$f_4(x) = \sum_{i=1}^{n} (\frac{10^6 (i-1)}{n-1}) x_i^2$	0	[-100,100]	Ellipsoidal
$f_5(x) = \sum_{i=1}^{n} i x_i^4 + random[0,1)$	0	[-1.28,1.28]	Noise
$f_6(x) = \sum_{j=1}^{n} (\sum_{i=1}^{j} x_i)^2$	0	[-100,100]	Quadric
$f_7(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	0	[-10,10]	Rosenbrock
$f_8(x) = \sum_{i=1}^{n} (floor(x_i + 0.5))^2$	0	[-100,100]	Step
$f_9(x) = \sum_{i=1}^{n} [x_i^2 - 10 \cos 2\pi x_i + 10]$*	0	[-5.12,5.12]	Rastrigin
$f_{10}(x) = -20 \exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos 2\pi x_i\right) + 20 + e$	0	[-32,32]	Ackley
$f_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(x_i/\sqrt{i}) + 1$	0	[-600,600]	Griewank
$f_{12}(x) = \sum_{i=1}^{n} [x_i^2 - 10 \cos 2\pi x_i + 10], x_i = \frac{round(2x_i)}{2} if \lvert x_i\rvert > 0.5$	0	[-5.12,5.12]	Non-continuous Rastrigin
$f_{13}(x) = \sum_{i=1}^{n} [y_i^2 - 10\cos(2\pi y_i) + 10]$; $y' = M \times x$; M is an orthogonal matrix	0	[-5.12,5.12]	Rotated Rastrigin
$f_{14}(x) = -20 exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} y_i^2}\right) - exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi y_i)\right) + 20 + e, y' = M \times x$; M is an orthogonal matrix	0	[-32,32]	Rotated Ackley
$f_{15} = \frac{1}{4000}\sum_{i=1}^{n} y_i^2 - \prod_{i=1}^{n} \cos(y_i/\sqrt{i}) + 1, y' = M \times x$;M is an orthogonal matrix	0	[-600,600]	Rotated Griewank
$f_{16} = 10^6 z_1^2 + \sum_{i=1}^{n} z_i^2 + 200, y' = M \times x$;M is an orthogonal matrix	200	[-100,100]	Rotated Kaatsura [18]
$f_{17}(x) = \frac{10}{n^2}\prod_{i=1}^{n}(1 + i\sum_{j=1}^{32} \frac{\lvert 2^j z_i - round(2^j z_i)\rvert}{2^j})^{\frac{10}{n^{1.2}}} - \frac{10}{n^2} - 1100, z = M_2 \Delta^{100}(M_1 \frac{5(x-o)}{100})$	-1100	[-100,100]	Rotated Discus [18]

TABLE III: Search result comparisons showing mean error, least error, standard deviation, rank according to the least mean error over 20 runs, 30 and 50 dimensions and 20000 Function Evaluation as terminating criterion

		30 dimension						50 dimension					
		MDO	DE	PSO	GSA	CLPSO	ABC	MDO	DE	PSO	GSA	CLPSO	ABC
f_1	Mean:	**0**	0.0137	0.0127	2.2e-15	0.6646	0.0072	**0**	19.793	50.610	1.4892	76.692	18.047
	Best:	**0**	0.0034	3.77e-4	4.0e-16	0.2591	0.0021	**0**	10.584	9.0843	2.39e-8	60.711	0.5451
	Dev:	**0**	0.0044	0.0077	1.0e-15	0.3233	0.0083	**0**	3.7979	13.788	1.3237	15.974	26.003
	Rank:	**1**	4	3	2	6	5	**1**	4	5	2	6	3
f_2	Mean:	**0**	4.5457	1.2747	0.5194	0.1266	0.0274	**0**	44.5795	269.117	1.8907	2.2078	0.5495
	Best:	**0**	0.7932	0.0315	1.21e-7	0.0837	0.0158	**0**	16.9204	41.9478	0.2207	2.0012	0.2897
	Dev:	**0**	1.3479	0.9249	0.4646	0.0353	0.0155	**0**	9.9817	115.712	0.7881	0.1689	0.2853
	Rank:	**1**	6	5	4	3	2	**1**	5	6	3	4	2
f_3	Mean:	**0**	1.85e+04	638.812	165.86	5.72e+05	6.50e+08	**0**	2.18e+07	3.97e+07	4.42e+5	6.96e+7	8.23e+08
	Best:	**0**	4.05e+03	519.978	28.871	3.14e+05	2.60e+06	**0**	1.75e+07	4.91e+06	59.5282	4.89e+7	1.53e+08
	Dev:	**0**	1.14e+04	51.292	91.808	1.7e+05	1.44e+09	**0**	2.61e+06	1.71e+07	3.63e+5	2.49e+7	4.84e+08

	Rank:	**1**	4	3	2	5	6	**1**	3	4	2	5	6
f_4	Mean:	**0**	5.66e+03	3.87e+03	2.5e-10	6.99e+04	56.032	**0**	5.54e+11	4.17e+10	8.67e+6	7.77e+11	1.8e+5
	Best:	**0**	3.02e+03	183.366	1.0e-10	3.68e+04	9.3818	**0**	3.02e+11	3.55e+09	6.82e+5	2.12e+11	4.5e+3
	Dev:	**0**	613.185	2.26e+03	5.1e-11	2.77e+04	35.815	**0**	1.10e+11	1.70e+10	4.26e+6	8.15e+11	2.6e+5
	Rank:	**1**	5	4	2	6	3	**1**	5	4	3	6	2
f_5	Mean:	**8.66e-6**	0.0527	0.2220	0.1293	0.0618	0.5371	**2.79e-5**	0.4459	87.179	0.5437	0.2494	1.6539
	Best:	**2.70e-6**	0.0425	0.1179	0.0866	0.0405	0.2926	**3.17e-6**	0.2247	0.9192	0.3406	0.1951	1.4271
	Dev:	**1.52e-6**	0.0042	0.0423	0.0168	0.0164	0.1912	**1.30e-5**	0.0746	60.354	0.1123	0.0756	0.1861
	Rank:	**1**	2	5	4	3	6	**1**	3	6	4	2	5
f_6	Mean:	**0**	1.22e+04	4.41e+03	652.134	1.06e+04	2.71e+04	**0**	1.25e+05	5.98e+04	3.55e+3	4.1006	7.8e+4
	Best:	**0**	5.85e+03	2.83e+03	422.694	5.83e+03	1.81e+04	**0**	7.94e+04	3.71e+04	1.72e+3	3.0262	6.2e+4
	Dev:	**0**	2.75e+03	429.0213	77.9220	2.76e+03	0.1912	**0**	1.27e+04	7.76e+03	526.237	8.3363	1.0e+4
	Rank:	**1**	5	3	2	4	6	**1**	6	4	3	2	5
f_7	Mean:	**24.970**	41.440	123.823	70.0148	284.487	94.452	**48.970**	1.29e+04	4.19e+04	515.193	2.23e+03	564.42
	Best:	**24.918**	29.170	73.8645	26.0692	246.818	21.661	**48.939**	955.88	803.306	185.652	1.74e+03	257.64
	Dev:	**0.0116**	10.126	19.2478	16.1135	31.7171	67.074	**0.0037**	1.01e+04	2.62e+04	199.220	471.425	579.49
	Rank:	**1**	2	5	3	6	4	**1**	5	6	2	4	3
f_8	Mean:	**0**	0.2000	0.8000	3.8000	1.8000	7.2	**0**	346	73	68.200	52	27
	Best:	**0**	**0**	**0**	**0**	**0**	3	**0**	16	46	45	41	9
	Dev:	**0**	0.1789	0.1789	2.7481	0.7376	4.5497	**0**	288.46	8.7040	11.2939	7.3144	18.165
	Rank:	**1**[#]	2	3	5	4	6	**1**	6	5	4	3	2
f_9	Mean:	**0**	192.660	69.6346	35.8185	12.1735	16.4621	**0**	414.736	223.913	89.210	58.6170	58.064
	Best:	**0**	158.860	49.2284	24.8740	10.2086	13.729	**0**	388.897	156.883	63.677	54.7209	47.824
	Dev:	**0**	10.7399	4.9990	4.2118	1.5127	3.2542	**0**	11.488	18.5964	7.568	3.1408	11.157
	Rank:	**1**	6	5	4	2	3	**1**	6	5	4	3	2
f_{10}	Mean:	**8.88e-16**	0.0321	2.1597	1.7e-8	1.3323	2.4341	**8.88e-16**	1.2509	13.935	0.9331	4.7550	6.2464
	Best:	**8.88e-16**	0.0167	1.1590	1.5e-8	0.9905	1.3082	**8.88e-16**	1.1442	7.1668	0.0032	4.5040	4.6229
	Dev:	**0**	0.0035	0.3295	8.0e-10	0.3444	0.7819	**0**	0.0448	2.4572	0.2199	0.2558	1.8016
	Rank:	**1**	3	5	2	4	6	**1**	3	6	2	4	5
f_{11}	Mean:	**0**	0.0976	0.0401	16.2887	0.8031	0.2140	**0**	1.3380	4.7489	48.149	1.7397	5.2540
	Best:	**0**	0.0237	0.0174	7.2883	0.6100	0.0423	**0**	1.1426	1.0738	39.925	1.5410	1.0167
	Dev:	**0**	0.0353	0.0089	3.5165	0.1530	0.2168	**0**	0.0689	1.9694	3.2583	0.1182	9.4017
	Rank:	**1**	3	2	6	5	4	**1**	2	4	6	3	5
f_{12}	Mean:	**0**	167.706	64.8960	49.4000	17.4632	18.1181	**0**	337.267	315.631	96.250	52.3041	45.761
	Best:	**0**	129.286	54.0914	31	15.7706	13.7515	**0**	301.667	236.027	73	45.6466	35.064
	Dev:	**0**	10.685	3.9686	4.2389	1.2150	7.1850	**0**	10.1028	22.9662	8.995	4.9856	8.2776
	Rank:	**1**	6	5	4	2	3	**1**	6	5	4	3	2
f_{13}	Mean:	**0**	247.521	210.3613	53.7277	237.709	422.454	**0**	480.035	597.336	101.88	481.721	910.64
	Best:	**0**	241.188	197.3677	42.7832	224.334	342.335	**0**	471.651	486.337	84.571	447.062	822.45
	Dev:	**0**	1.6668	4.7753	4.6241	14.4474	68.5287	**0**	3.3822	36.6437	7.031	26.3411	116.38
	Rank:	**1**	5	3	2	4	6	**1**	3	5	2	4	6
f_{14}	Mean:	**0**	21.0253	21.0463	20.2871	21.0840	21.1479	**0**	21.1697	21.2058	20.297	21.2121	21.287
	Best:	**0**	20.9142	20.9710	20.1896	20.9947	21.043	**0**	21.0937	21.0900	20.206	21.1564	21.254
	Dev:	**0**	0.0364	0.0217	0.0425	0.0616	0.0595	**0**	0.0184	0.0310	0.0265	0.0435	0.0280
	Rank:	**1**	3	4	2	5	6	**1**	3	4	2	5	6
f_{15}	Mean:	**0**	1.0777	1.0611	31.6901	7.3881	4.4818	**0**	8.7673	20.4163	229.50	41.4544	38.7855
	Best:	**0**	0.9712	1.0299	20.0152	5.8486	1.9085	**0**	3.2269	15.1175	180.99	38.1179	11.132
	Dev:	**0**	0.0340	0.0160	4.7340	1.2374	2.8984	**0**	2.5646	2.4401	17.967	3.9588	31.494
	Rank:	**1**	3	2	6	5	4	**1**	2	3	6	5	4
f_{16}	Mean:	**0.2322**	0.4059	2.1324	0.2333	0.4072	0.4060	**0.3984**	0.4059	0.405	0.417	0.4890	0.432
	Best:	**0.2322**	0.4055	2.0935	0.2323	0.4002	0.4050	**0.3985**	0.4055	0.403	0.414	0.4310	0.431
	Dev:	**0**	5.34e-3	0.0352	0.0021	0.0043	8.67e-04	**4.2e-04**	5.34e-4	2.12e-3	0.0031	1.4e-02	1.96e-4
	Rank:	**1**	3	6	2	5	4	**1**	3	2	4	6	5
f_{17}	Mean:	1.27e+5	5.62e+5	**1.03e+5**	5.17e+6	1.12e+5	3.10e+5	2.31e+5	3.70e+5	**2.02e+5**	2.78e+8	2.38e+5	5.40e+5
	Best:	1.11e+5	4.15e+5	**8.80e+4**	5.14e+6	8.1e+4	2.39e+5	2.18e+5	7.44e+5	**1.60e+5**	2.77e+8	2.21e+5	4.91e+5
	Dev:	153.2	2.27e+5	**1.84e+4**	2.16e+5	8.3421	4.91e+4	1.41e+1	3.17e+5	**4.54e+4**	1.2e+05	0.4521	5.10e+4
	Rank:	3	5	**1**	6	2	4	2	5	**1**	4	3	6

IV. Parameter Tuning

The three parameters of (6), which are c_1, c_2 and c_3, are varied to show the robustness of the proposed technique. Willpower coefficient c_1 should lie between (0, 1) since an individual's thought process can be partially influenced by his own previous thought process and thus never exceed 1. Value of 0 indicates that the individual starts thinking afresh. Value of 1 indicates that the individual adopts the parameter blindly. The extreme values may not be conducive for the society as there is a high probability that the individuals will be misled if the parameter weight is made 0 and there may be a lack of rationality if the individual blindly follows any one of the three parameters, monarch's thought, the peak thought of his lifetime or his thought at the previous iteration. Similar reasoning for past experience coefficient c_2 and commanding coefficient c_3 makes the values of both c_2 and c_3 lie between (0, 1). We vary the parameters over a range of (0, 1) by distributing them equally over the entire range - 0.2, 0.4, 0.6 and 0.8.

Fig. 2 is obtained by running MDO technique for each function once over 20 citizens, 30 dimensions and 20000 FEs as the terminating criteria. From fig. 2, it is found that the MDO technique is insensitive to parameter tuning and hence more robust as all the values tested on unimodal, multimodal and rotated functions show that it gives the same global optima. However, it is empirically found that lower the value of the three coefficients, steeper is the convergence curve and more efficient is the technique as it requires less number of Function Evaluations (FEs) to converge to its optimum value. Despite being independent of parameter tuning, it has been found empirically found that the optimum values of c_1, c_2 and c_3 as 0.2, 0.2 and 0.2 respectively for faster convergence. An important observation is that if any of the three parameters are increased or given more weight than the other parameters, then the number of FEs increases. This may be attributed to the fact that the individual's previous thought may saturate him, or the best thought process of an individual's lifetime or the influence of the monarch's thoughts may not be conducive for the betterment of the society thus stagnating it.

Table IV and V give the mean error value and corresponding standard deviation over a period of 20 runs for 30 and 50 dimensions respectively after tuning the individual parameters c_1, c_2 and c_3 over (0,1), with 20000 FEs as terminating criteria. We find that the proposed technique is parameter insensitive as the standard deviation and mean error remain more or less unaltered even when the parameters are tuned, which gives it robustness.

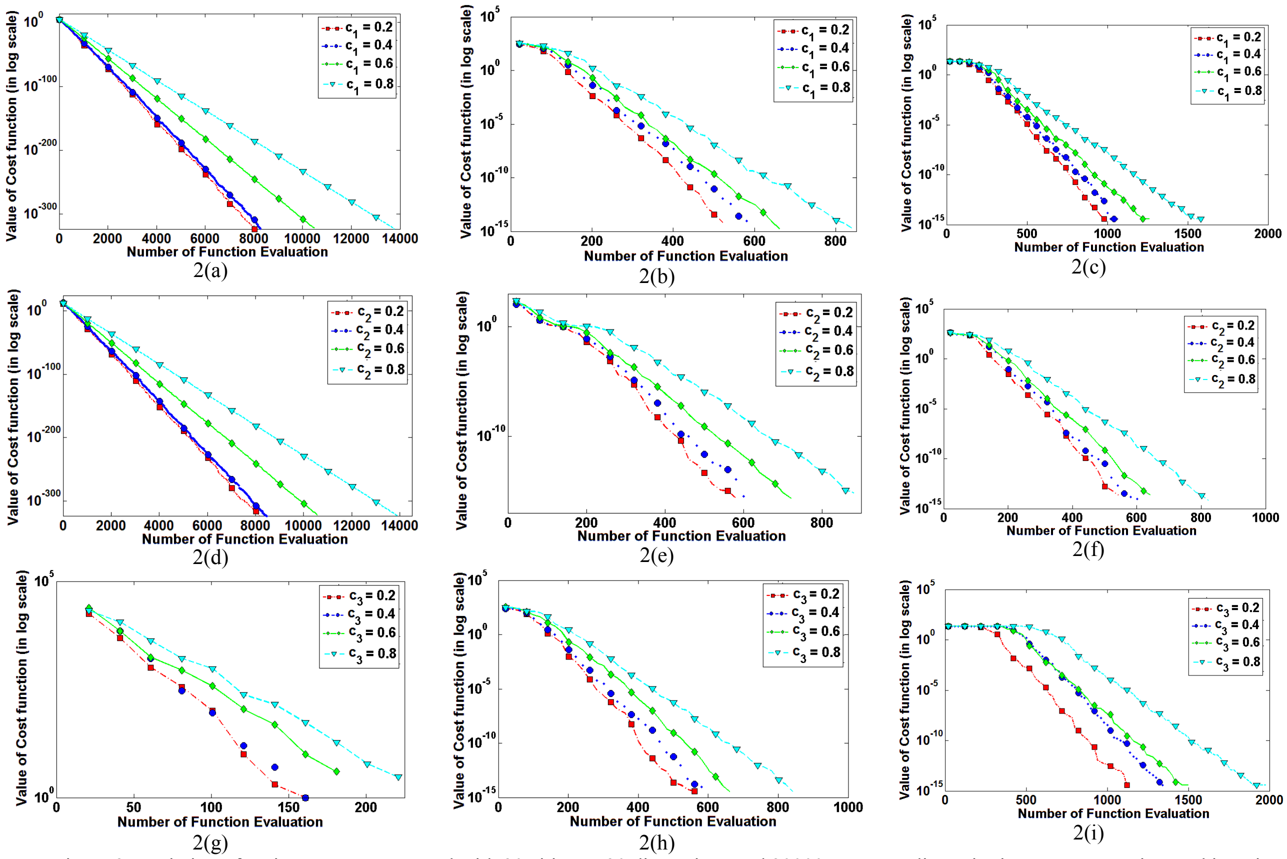

Figure 2: Variation of tuning parameters tested with 20 citizens, 30 dimensions and 20000 FEs as ending criteria. Parameter c_1 is tuned in unimodal Sphere function, multi-modal Rastrigin and rotated Ackley in Fig 2(a), (b) and (c) respectively. Parameter c_2 is tuned in unimodal Ellipsoidal function, multi-modal Griewank and rotated Rastrigin in Fig 2(d), (e) and (f) respectively. Parameter c_3 is tuned in unimodal Step function, multi-modal non-continuous Rastrigin and rotated Griewank in Fig 2(g), (h) and (i) respectively.

TABLE IV: Mean error (Standard Deviation) with 20 citizens, 30 dimensions and 20000 FEs as terminating criteria over the benchmark functions f_1 – f_{17} by varying the tuning parameters c_1, c_2 and c_3 over a period of 20 runs from 0 to 1

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}
c_1	0 (0)	0 (0)	0 (0)	0 (0)	8.25e-6(0)	0 (0)	24.91	0 (0)	0 (0)	8.88e-16 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0.23(0)	1.28e+5 (0.18)
c_2	0 (0)	0 (0)	0 (0)	0 (0)	8.2e-6(0)	0 (0)	24.31	0 (0)	0 (0)	8.88e-16 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0.24(0)	1.31e+5 (0.24)
c_3	0 (0)	0 (0)	0 (0)	0 (0)	8.17e-6(0)	0 (0)	24.73	0 (0)	0 (0)	8.88e-16 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0.34(0)	1.32e+5 (0.22)

TABLE V: Mean error (Standard Deviation) with 20 citizens, 50 dimensions and 20000 FEs as terminating criteria over the benchmark functions $f_1 - f_{17}$ by varying the tuning parameters c_1, $c_2$and c_3 over a period of 20 runs from 0 to 1

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}
c_1	0 (0)	0 (0)	0 (0)	0 (0)	2.78e-5	0 (0)	48.37	0 (0)	0 (0)	8.8e-16(0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0.39(4.1e-4)	2.33e+5 (13.1)
c_2	0 (0)	0 (0)	0 (0)	0 (0)	2.86e-5	0 (0)	48.21	0 (0)	0 (0)	8.8e-16(0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0.44(5.1e-3)	2.42e+5 (12.4)
c_3	0 (0)	0 (0)	0 (0)	0 (0)	2.94e-5	0 (0)	48.33	0 (0)	0 (0)	8.8e-16(0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0.45(6.3e-3)	2.47e+5 (15.2)

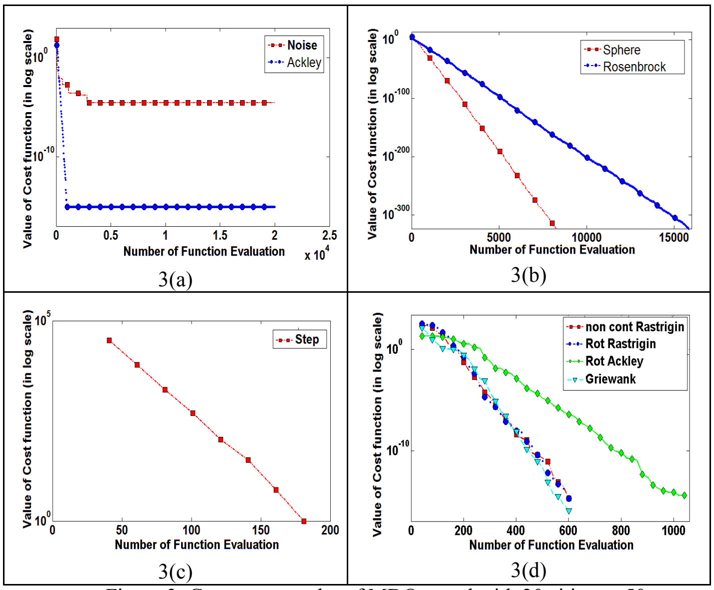

Figure 3: Convergence plot of MDO tested with 20 citizens, 50 dimensions and 20000 FEs as ending criteria.

V. Conclusion

This paper introduces Monarchy Driven Optimization Technique (MDO), a new kind of human society inspired optimization technique. The equations governing the MDO come from the society ruled by monarchs. The monarch rules the kingdom by his thoughts which become manifested in his outlook and can be evaluated from the cost functions. Meanwhile the citizens also contribute to the society by getting influenced by themselves and the monarch, and the monarch also values their thoughts by getting influenced by the best thought among the citizens. Tested on unimodal and multimodal benchmark functions and compared with other well known variants of optimization techniques like DE, PSO, CLPSO, ABC and GSA, MDO is seen to be efficient for unimodal, multimodal and complex multimodal functions, and also capable of overcoming premature convergence. Also 20000 Function Evaluations are chosen to show the fast converging nature of the technique. The simple nature of the technique is an added advantage. Another important aspect is that it is parameter independent thus making it a robust one.

For future research work, the technique can be tested and improved for multi-objective functions. It remains to be seen how the technique can perform in problems involving applications such as electromagnetism and image processing where numerous local minima and complex nature of the functions hamper the process of optimization.

Reference

[1] J. F. Frenzel, "Genetic algorithms: A new breed of optimization", IEEE Potentials, vol. 12, pp.21 -24, 1993.

[2] J. Kennedy, and R. C. Eberhart, "Particle swarm optimization ". Proceedings of IEEE International Conference on Neural Networks, pp. 1942-1948, 1995.

[3] R. C. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory,” in Proceedings of 6th International Symposium on Micromachine Human Science, 1995, pp. 39– 43.

[4] K. Socha, M. Dorigo, “Ant colony optimization for continuous domains”, European Journal of Operational Research 185(3): 1155-1173 (2008).

[5] T. Liao, M. A. Montes de Oca, D. Aydin, T. Stutzle, M. Dorigo, “An incremental ant colony algorithm with local search for continuous optimization”, GECCO 2011, pp. 125-132.

[6] T. Liao, T. Stutzle, M. A. Montes de Oca, M. Dorigo, “A unified ant colony optimization algorithm for continuous optimization”, European Journal of Operational Research, 234(3): 597-609, 2014.

[7] K. V. Price and R. Storn, "Differential evolution: A simple evolution strategy for fast optimization", Jounaal of Global Optimization, vol. 22, no. 4, pp.18 -24, 1997.

[8] R. Storn and K. Price, "Differential evolution- A simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization, vol. 11, no. 4, pp. 341-359, 1997.

[9] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art," IEEE Transactions on Evolutionary Computation, vol. 15, no. 1, pp. 4-31, 2011.

[10] D. Karaboga, B. Akay; “A survey: algorithms simulating bee swarm intelligence”, Artificial Intelligence Review; 31 (1), pp. 68-85, 2009.

[11] J. J. Liang, A. K. Qin and P. N. Suganthan, “Comprehensive Learning Particle Swarm Optimiser for Global Optimisation of Multimodal Functions”,IEEE Transactions on Evolutionary Computation, vol. 10, no. 3, pp. 281-295, June,2006.

[12] K. Diwold, M. Beekman and M. Middendorf, “Honeybee optimisation – an overview and a new bee inspired optimisation scheme”, Handbook of Swarm Intelligence Adaptation, Learning and Optimisation, vol. 8, pp. 295-327, 2010.

[13] A. Gargari and C. Lucas, "Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition", Evolutionary Computation, CEC 2007. IEEE Congress on, 2007, pp. 4661-4667.

[14] E. Rashedi , H. Nezamabadi-pour and S. Saryazdi "GSA: A gravitational search algorithm", Information Sciences, vol. 179, pp.2232 -2248, 2009.

[15] Z. Bayraktar, M. Komurcu, and D. H. Werner, "Wind driven optimization (WDO): A novel nature-inspired optimization algorithm and its application to electromagnetics”, Proceedings of the 2010 IEEE International Symposium on Antennas and Propagation and CNC/USNC/URSI Radio Science Meeting, Toronto, Ontario, Canada, July 11-17, 2010.

[16] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization", IEEE Transaction on Evolutionary Computation, vol. 1, pp.67 -82, 1997.

[17] Y. Shi, “Brain storm optimization algorithm”, ICSI 2011, Part I, LNCS 6728, pp. 303–309, 2011.

[18] J. J. Liang, *et al*., "Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization”, Com. Intel lab., Zhengzhou University, Zhengzhou, China, Tech. Rep. and Nanyang Technological University, Singapore, Tech. Rep., 2013.