A New Global Optimization Algorithm Inspired by Parliamentary Political Competitions

Ali Borji

School of Cognitive Sciences,
Institute for Studies in Theoretical Physics and Mathematics, Tehran, Iran
`borji@ipm.ir`

Abstract. A new numerical optimization algorithm inspired by political competitions during parliamentary head elections is proposed in this paper. Competitive behaviors could be observed in many aspects of human social life. Our proposed algorithm is a stochastic, iterative and population-based global optimization technique like genetic algorithms and particle swarm optimizations. Particularly, our method tries to simulate the intra and inter group competitions in trying to take the control of the parliament. Performance of this method for function optimization over some benchmark multi-dimensional functions, of which global and local minimums are known, is compared with traditional genetic algorithms.

Keywords: Function Optimization, Politics, Political Competitions, Evolutionary Algorithms, Political Optimization, Stochastic Optimization.

1 Introduction

Global optimization is the task of finding a solution set of an objective function in which it takes its smallest value, the global minimum. From a computational complexity point of view, finding the global minimum of a function is a NP-hard problem, which means that there is in general no algorithm for solving it in polynomial time. Global optimization is a stronger version of the local optimization, in sense that instead of searching for a locally improvable feasible point, it looks for the best point in the search space. In many practical engineering applications finding the globally best point is desirable but not critical, since any sufficiently suboptimal solution which satisfies a specific criterion is acceptable. It is also possible to manually improve it further without optimization in some cases. For such problems there is a little harm in doing an incomplete search [1].

Researchers in different fields have got many inspirations from the solutions that nature has evolved for hard problems. An interesting such areas is optimization. Taking advantage of evolutionary approach nature has selected, genetic algorithms and its variants has been suggested for optimization. Modeling animal behaviors has resulted to particle swarm [2] and ant colony optimization algorithms [3]. Recently simulating social behaviors of humans has leaded to some solutions in engineering [4], [5].

Our aim in this paper is to introduce a new optimization method inspired from human social behavior in political situations. As we are going to compare our method with genetic algorithms a brief introduction to it is followed.

Genetic algorithm, GA, was first introduced by Holland [6]. It takes the essence of biological evolution by activating mutation and crossing-over among candidates. GA is a stochastic, incomplete, iterative, and population-based global optimization method. In each iteration of GA, a competitive selection weeds out poor solutions. Solutions with high "fitness" are then "recombined" with other solutions by swapping their parts. A "mutation" operation is also applied over solutions by making a small change at a single element. Recombination and mutation are used to generate new solutions that are biased towards regions of the space for which good solutions have already been experienced.

Competitions among political groups during head elections in a parliament have been our main inspiration source for formulating a new method for global optimization in this paper. All individuals of the population are clustered randomly into some groups. Members of a party fall into two categories: regular members and candidate members. Parliamentary head candidates compete in each group to become the final candidate of that group. A final candidate then must compete in next round with candidates of other groups for parliament head position. Intra-party and inter-party competitions guide the algorithm to converge to the global minimum of the objective function. We have overridden some real life details purposely for simplicity. For example in parliamentary system of some countries people vote for candidates both within a group and among groups. This algorithm has some similarities with competitions during athletic championships in which athletics are grouped into teams and they play against each other to ascend from their teams and then play with winners of other teams in the next round. But obviously our method has certain differences. As it can be seen it is also somehow similar to competitions among parties during presidential campaign.

The reminder of this paper is organized as follows. In section two a brief introduction to the political competitions to the extent relevant to our work is described. Details of the proposed optimization method are introduced in section three. It is followed by simulation studies in section four. Finally section five draws conclusions and summarizes the paper.

2   Parliamentary Political Competitions

It is a common incident and is repeatedly observed in different aspects of human life that people tend to form social groups. In sociology, a group is usually defined as a collection of humans or animals, who share certain characteristics, interact with one another, accept expectations and obligations as members of the group, and share a common identity. Using this definition, society appears as a large group. Characteristics that members in the group may share include, interests, values, ethnic and linguistic background, and kinship ties. Two types of relationships exist within a group and among groups: competition and cooperation. Members of a group (human social group) have different capabilities making each one suitable for a certain task. Because each member has a collection of needs and capabilities, it forms a potential

for them to cooperate. At the same time they compete to earn higher fraction of group resources. In competition among groups, they compete to get better situations and take the superiority over others to attain the limited sources of their living environment. Although many patterns of such behaviors are observable in human social life, we constrain ourselves to a specific competition-cooperation behavior during parliamentary elections.

A parliamentary system, also known as parliamentarianism is a governmental system in which the power to make and execute laws is held by a parliament. Members of the parliament are elected in general elections by people. People usually vote in favor of parties. Members of a parliament belong to political parties. They support their parties in parliamentary votes. Clustering members of the parliament into clusters, based on the party they belong, results to competitions among parties in trying to gain superiority over other parties. Almost in all democratic countries, political parties form the population of parliaments [7].

There are basically two systems in parliamentary elections: the Majority Election System and the Proportional Representation System. In the majority election system, only one Member of Parliament is elected per constituency. In the proportional representation system several members of parliament are elected per constituency. Basically every political party presents a list of candidates and voters can select a list that is they vote for a political party. Parties are assigned parliamentary seats proportionally to the number of votes they get.

Political parties, either in the parliament or out of it, have members with different levels of power. Those main people of a party try to make good impacts over other regular members with less power. They do that to benefit from their support and votes during elections, etc. Therefore, important members of parties are engaged in competitions and try to find supporters among regular members. On the other hand, regular members have tendency toward more capable persons and usually vote for people they believe in. In this dynamic process, regular members with high capability replace previous candidates. These competitions are among individuals within parties. Another kind of competition is at the level of parties. Political parties compete for gaining more power. Two main goals that parties try to achieve are greater number seats in the parliament and taking the control of government.

3 Parliamentary Optimization Algorithm (POA)

Optimization process in our algorithm is started by first creating a population of individuals. These individuals are assumed to be the members of the parliament. In the next step, population is divided into some political groups (parties) and a fixed number of members with highest fitness are selected as group candidates (leaders).

After partitioning the population, intra-group competition is started. In intra-group competition a regular members get biased toward candidates in proportion to their fitness. It is motivated from the fact that a regular member is usually under impact of superior members. This observation is modeled here as a weighted average of vectors from a regular member to candidates. This causes the evolutionary process search for potential points in search space and provides an exploratory mechanism. At the end of intra-party competition a few candidates with highest fitness are regarded as final

candidates of the group. They compete with candidates of other groups in next stage. Both candidates and regular members of a group are important in determining total power of a group. A linear combination of mean power of candidates and mean power of regular members is considered as the total fitness of a group. As in parliamentary system of some countries no voting mechanism is assumed. Actually, the biasness mechanism could somehow be considered as implicit voting.

Inter-group competition begins just after intra-group competitions ends. Political groups within the parliament perform competition with other groups to impose them their own candidate. In our method, the role of groups is still preserved after introducing a candidate. Each group not being able to compete with others becomes weaker and loses its chance to take the parliament head position.

Groups with a negligible fitness gradually lose their power and ultimately collapse. On the other hand, stronger groups become progressively more powerful and consequently earn higher chance to win the competition. Powerful groups sometimes agree to join and merge into one (at least at on some special occasions) to increase their wining chance. This gives the search mechanism chance to look on more promising areas therefore offers a mechanism for exploitation. The tendency of regular members of a group toward their group candidates along with affinity of powerful groups to join and also collapse mechanism drives convergence to a state in which there exists just one group in the parliament. In contrast to what happens in real world, when algorithm converges, regular members have near the same or equal power as the candidate which is now the head. A step by step description of the algorithm is summarized as follows:

1. Initialize the population.
2. Partition population into some groups.
    2.1. Pick θ highly fitted individuals as candidates of each group.
3. Intra-group competition
    3.1. Bias regular members toward candidates of the group.
    3.2. Reassign new candidates.
    3.3. Compute power of each group.
4. Inter-group competition
    4.1. Pick λ most powerful groups and merge them with probability P_m.
    4.2. Remove γ weak groups with probability P_d.
5. If stopping condition not met go to 3.
6. Report the best candidate as the solution of the optimization problem.

3.1 Population Initialization

A population of initial solutions with size N is being dispread over the d-dimensional problem space at random positions. Each individual of the population is coded as a d-dimensional continuous vector:

$$P = [p_1, p_2, \dots , p_d], p_i \in \text{IR} \tag{1}$$

Each individual could be either a regular member or candidate of a given group. A fitness function f is used to calculate the strength of an individual.

3.2 Population Partitioning

In order to form initial groups, population is portioned into M divisions. Each group contains L individuals. N, M and L are positive integers and are selected in such a way to satisfy the following equation:

$$N = M \times L \tag{2}$$

Top $\theta < L/3$ candidates with high fitness are then considered as candidates of each group. At this point all groups have the same number of members, but in the course of running the algorithm groups might earn different number of individuals because of merge and collapse mechanisms.

3.3 Intra-group Competition

Regular members of a group get biased toward candidates after interactions take place between candidates and regular members. This biasness is assumed here to be linearly proportional to weighted average of vectors connecting a member to candidates. Each candidate is weighted to the extent of its candidate fitness as shown in equation 3.

$$p' = p_0 + \eta \left(\frac{(p_1 - p_0) \cdot f(p_1) + (p_2 - p_0) \cdot f(p_2) + (p_3 - p_0) \cdot f(p_3)}{f(p_1) + f(p_2) + f(p_3)} \right) \tag{3}$$

In above formula, η is a random number between 0.5 and 2 and allows the algorithm to search in a local search area around candidates. Another alternative mechanism is to use large values of η at first iterations and the gradually reduce it, perhaps by analyzing variance. Fig. 1 illustrates biasing mechanism.
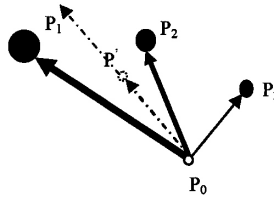

Fig. 1. Biasing a member toward candidates

A regular member is allowed to change, only if it takes a higher fitness value. After biasing, regular members might have higher fitness values than candidates. In such cases, a reassignment of candidates is done. Let $\underline{Q}^i = [Q_1, Q_2, \dots, Q_\theta]$ be the vector of candidates and $\underline{R}^i = [\underline{R}_{\theta+1}, \underline{R}_{\theta+2}, \dots, \underline{R}_L]$ the remaining regular members of the i-th group, power of this group is calculates as:

$$Power^i = \frac{m \times Avg(\underline{Q}^i) + n \times Avg(\underline{R}^i)}{m+n}; \ m > n \tag{4}$$

3.4 Inter-group Competition

Stronger groups sometimes, join and merge to one group in order to amplify their power. To perform merging, a random number is generated and if it is smaller than P_m, λ most powerful groups are picked and merged into one. During the course of running algorithm, weak groups are removed to save computation power and reduce function evaluations. Like merging, a random number is generated and if it is smaller than P_d, γ groups with minimum power are eliminated.
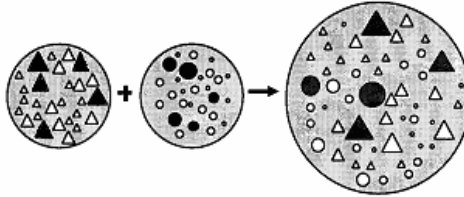

Fig. 2. Merging two groups into one group

3.5 Stopping Condition

At the end of algorithm, a group wins the competitions and its best member (candidate with maximum fitness) in considered as the solution of the optimization problem. Two stopping conditions are possible. Algorithm terminates if either maximum number of iterations is reached or during some successive iterations no significant enhancement in fitness is observed.

4   Simulation Studies

Several experiments are conducted in this section to demonstrate success of the proposed algorithm. Specifically, capability of the algorithm in finding global minimum of three benchmark functions 'Sphere', 'Rastrigin' and 'Ackley' is investigated. Plots of these functions in two dimensions are shown in Fig. 3. Efficiency of the parliamentary optimization algorithms is also compared with traditional genetic algorithm over these problems. To do experimentation with genetic algorithm, GA toolbox provided with MATLAB® was used. Table 1 shows POA parameter values for minimizing optmization function. In order to do a fair comparison with GA, initial conditions and number of initial individuals were identical in simulations.

4.1 Sphere Function

Equation five defines the sphere function in n dimensions. Dynamics of individual behaviors of the best group around the point with the best fitness is shown in top part of Fig 4. It can be observed that individuals move toward the optimal point from initialization area. In their progress towards the optimal point, individuals are biased toward the candidates of groups. This process gradually moves the points toward the
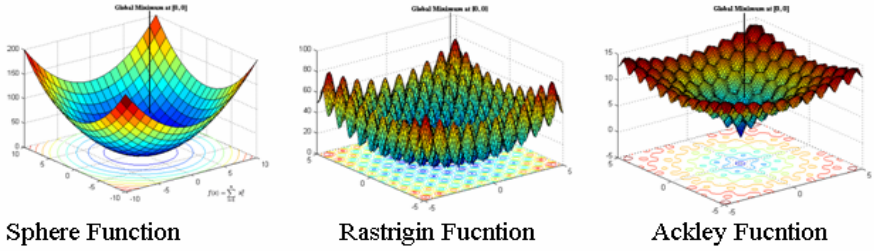
Fig. 3. Plot of functions used in simulations. Black vertical bar points to minimum.

Table 1. POA parameter values for mimization of three optimization problems

Parameter	Quantity	Sphere	Rastrigin	Ackley
N	Number of groups	5	6	10
L	Group size (individuals)	8	10	10
X_{ini}	Initial search area	$-40 < X_{ini} < -30$	$-20 < X_{ini} < -10$	$-20 < X_{ini} < -10$
P_m	Merge probability	0.01	0.05	0.01
P_d	Deletion probability	0	0.0025	0.002
θ	Number of candidates	2	3	2
d	Dimension	2	2	10
η	Biasness parameter	**0.5<random number < 2**		
m	Candidate weighting constant	1	1	1
n	Member weighting constant	0.01	0.01	0.01
λ	Groups to be merged	**2**		
γ	Groups to be deleted	**1**		
Itr	Maximum iterations	100	500	1000

global minimum. The minimum value of the sphere function discovered by POA was $y = 7.89$ e -11 at the point x = [–0.1413e–3, –0.0662e–3]. It is known that the optimal value of this function is zero for the point [0, 0] in the x-y plane. Minimum, maximum and average of individuals in population as well as fitness variance is plotted in Fig. 4. Compared with GA, our methods showed significance enhancement o 100 iterations.

$$f(x) = \sum_{i=1}^{n} x_i^2 \tag{5}$$

4.2 Rastrigin Function

We address another challenging optimization problem, which is minimization of Rastrigin function to demonstrate the effectiveness the POA. Fig. 3 clearly shows that the Rastrigin function has numerous local minima. However, it has just one global minimum, at the point [0, 0] in the x–y plane, as indicated by the vertical line in the plot, where the value of the function is zero. Rastrigin function is defined as below:

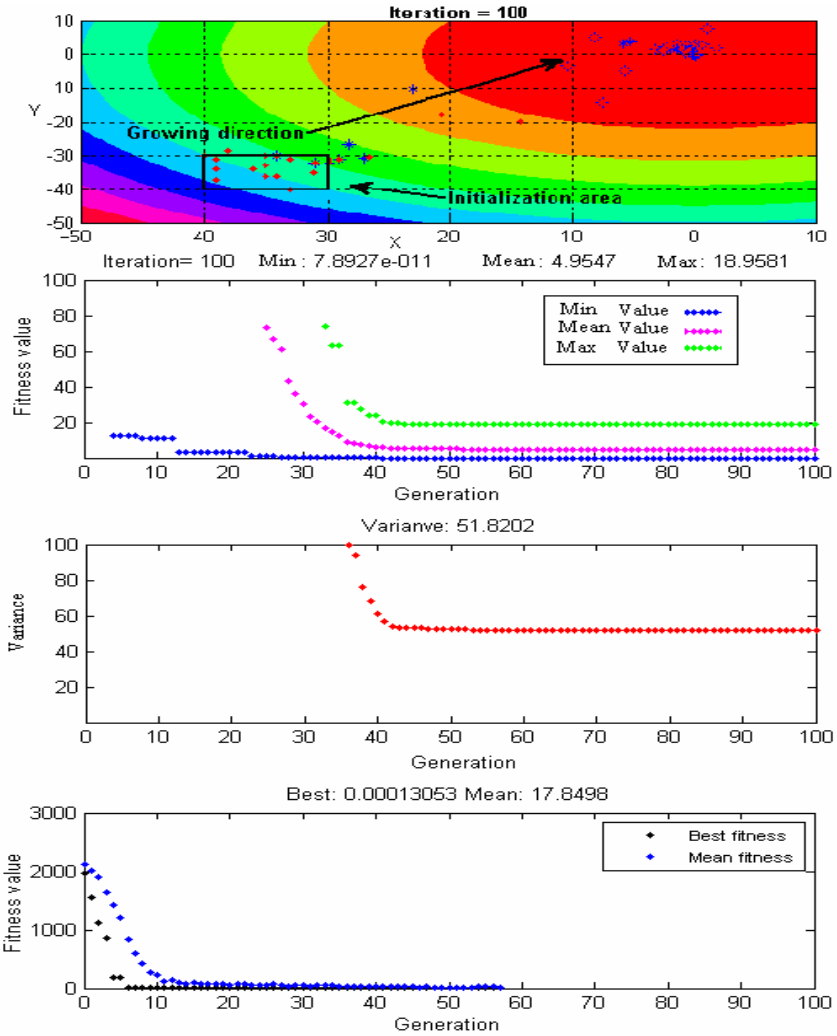$$f(x) = 10n + \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i)) \tag{6}$$

Fig. 4. From top to bottom: Convergence of the population toward the optimal point over Sphere function. Minimum, mean and maximum fitness are plotted for each generation over entire population. High variance in first iterations decreases at algorithm converges to the solution. Performance of GA over sphere function is significantly worse than POA.

Performance of POA and GA over Rastrigin function is shown in Fig. 5. POA reached absolute zero (in MATLAB) after 293 iterations. Algorithm stopped after no change was seen over fitness landscape. Again in comparison with GA a significant improvement was achieved.

4.3 Ackley Function

Ackley function is a challenging and favorite benchmark problem for optimization algorithms shown in equation 7. It could be observed from Fig. 3 that the function has

only one global minima at [0, 0] in x−y plane with also numerous local minima. In this experiment we aimed to compare the efficiency of POA with GA over a high dimensional optimization problem. Results over this function are illustrated in Fig. 6.

$$f(x) = 20 + e - 20\,e^{-\frac{1}{5}\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}} - e^{-\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)} \tag{7}$$
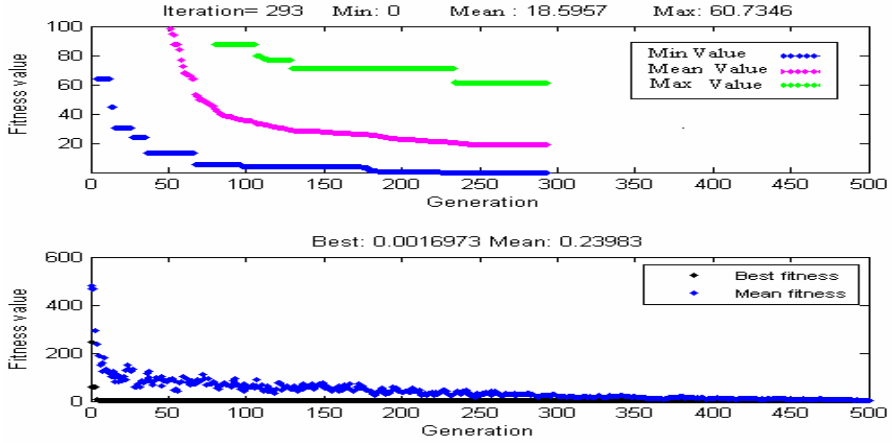

Fig. 5. Performance of POA and GA over Rastrigin function is shown in this figure. Note that GA has fall into local minima.
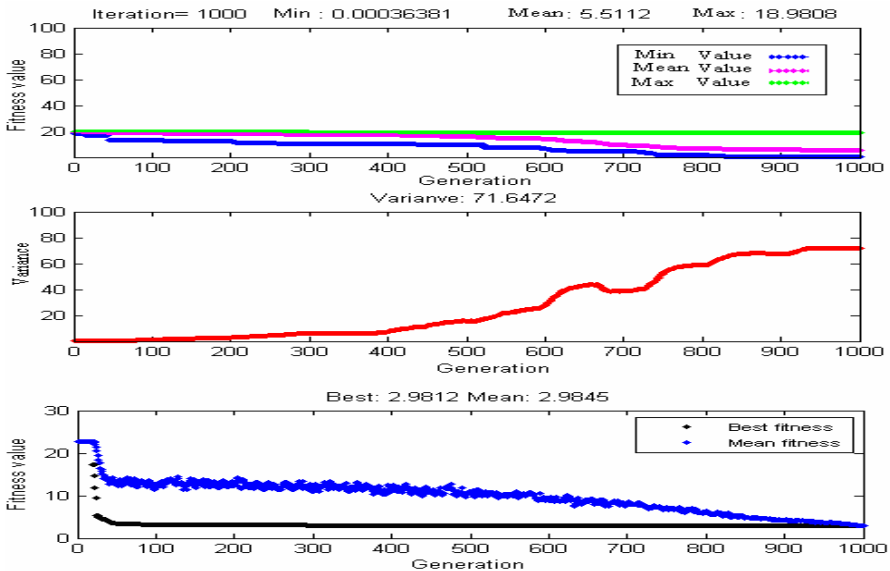

Fig. 6. POA is compared with GA over 10-dimensional Ackley function. Again GA has fall into local minima. Gradual increase in variance is because of local minima in function.

5 Conclusions

In this work, we introduced a new global optimization algorithm which is inspired from competitions among political parties trying to take the control of the parliament. Although we have bypassed some of the details of these competitions, proposed algorithm still has acceptable efficiency.

An initial population is created at the first step. It is then partitioned into some political groups. Each member of a given group is either a regular member or a superior member (candidate) of that group. Intra-group competition is the attempt of superior members to get the support of regular members. Regular members fluctuate with their bias toward superior members based on their achievements. In inter-group competition, political groups engage in competition and cooperation behaviors to win a good situation. This cooperation is modeled in this paper as merging those more powerful groups to one bigger more powerful one. Some weak groups which have no positive effect on search process are removed. That way, stronger groups become gradually more powerful while weaker ones become weaker and finally collapse.

At the end of these competitions, the most superior of the most powerful party becomes the leader or the head of the parliament and is considered as the solution of the optimization problem.

Several numerical simulations are carried out to investigate the convergence of POA over three benchmark optimization problems. Results show significant enhancement over staged genetic algorithm over three problems.

As it can be from simulations our proposed optimization algorithm captures important essences of political competitions fairly well and is capable to find desired minima very fast in comparison with other stochastic search algorithms. As an optimization algorithm, it has the additional desirable properties of capability to deal with complex and non-differentiable objective functions and escapes from local optima.

Through investigation of algorithm parameters, comparison with other optimization techniques like particle swarm and ant colony optimization as well as experimenting with a rich repertoire of high dimensional benchmark problems are the areas authors suggest for future works.

References

1. Horst, R., Pardalos, P.M., Thoai, N.V.: Introduction to Global Optimization, 2nd edn. Kluwer Academic Publishers, Dordrecht (2000)
2. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks, pp. 1942–1948. IEEE Computer Society Press, Los Alamitos (1995)
3. Dorigo, M., Maniezzo, V., Colorni, A.: The ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man and Cybernetics. Part B. Cybernetics 26(1), 1–13 (1996)

4. Hemingway, C.J.: Socio-cognitive Research: Toward a socio-cognitive theory of information systems: an analysis of key philosophical and conceptual issues. In: the Workshop Proceedings of Information Systems: Current Issues and Future Changes (1998)
5. Sharples, M.: Socio-cognitive engineering: a methodology for the design of humancentred technology. European Journal of Operational Research (2002)
6. Holland, J.H.: Adoption in Natural and Artificial Systems. University of Michigan, Ann Arbor (1975)
7. Shourie, A.: The parliamentary system. Rupa & Co, USA (2007)