Social Evolution: An Evolutionary Algorithm Inspired by Human Interactions

R. S. Pavithr and Gursaran

Abstract Inherent intelligent characteristics of humans, such as human interactions and information exchanges enable them to evolve more rapidly than any other species on the earth. Human interactions are generally selective and are free to explore randomly based on the individual bias. When the interactions are indecisive, individuals consult for second opinion to further evaluate the indecisive interaction before adopting the change to emerge and evolve. Inspired by such human properties, in this paper a novel social evolution (SE) algorithm is proposed and tested on four numerical test functions to ascertain the performance by comparing the results with the state-of-the-art soft computing techniques on standard performance metrics. The results indicate that, the performance of SE algorithm is better than or quite comparable to the state-of-the-art nature inspired algorithms.

Keywords Society and civilization · Social evolution · optimization

1 Introduction

Nature inspired computing is one of the main branches of natural computing techniques and an emerging computational paradigm for solving large-scale complex and dynamic real-world problems. Nature inspired computing builds on the principles of emergence, self-organization, and complex systems [1]. One of the main objectives of the nature inspired computing paradigm is to provide alternative stochastic, nature inspired search-based techniques to problems that have not been

R. S. Pavithr (✉) · Gursaran
Dayalbagh Educational Institute, Dayalbagh, Agra, India
e-mail: rspavithr@ieee.org

Gursaran
e-mail: gursaran.db@gmail.com

B. V. Babu et al. (eds.), *Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012), December 28–30, 2012*, Advances in Intelligent Systems and Computing 236, DOI: 10.1007/978-81-322-1602-5_153,

(satisfactorily) resolved by traditional deterministic algorithmic techniques, such as linear, nonlinear, and dynamic programming, etc [2].

Some of the interesting algorithms inspired by the natural phenomena are:

- Algorithms inspired by biology
- Algorithms Inspired by the behavior of groups of agents (Swarm)
- Algorithms inspired by human interactions and beliefs in the society.

For the past two decades, the research in the nature inspired evolutionary algorithms has been focused upon algorithms inspired by biological processes [3–7] and intelligent foraging behavior of social insects such as ants, birds, bees. For example, in the natural ant system, ants interact with each other indirectly by sharing the Pheromone trails. Swarm intelligence can be defined as "a property of a system of unintelligent agents of limited individual capabilities exhibiting collective intelligent behavior" [8]. Inspired by natural ant system, artificial ant colony optimization was designed and successfully implemented for many practical complex applications. Mimicking the behavior of flying birds, Kennedy and Ebergart [9] introduced another popular swarm-based evolutionary algorithm called particle swarm optimization. Artificial bee colony (ABC) algorithm is another stochastic search-based technique that belongs to class of swam intelligence-based algorithms, which is inspired by the intelligent behavior of the honeybees [10].

In 1994, Reynolds, designed a new algorithm called Cultural algorithm inspired by human interactions and beliefs and he argued that, "the cultural evolution enables the societies to evolve or adapt to their environments at rates that exceed that of biological evolution based on genetic inheritance only" [11]. Cultural algorithms are a class of computational models of cultural evolution that support dual inheritance, from belief space and individual interactions. This model of dual-inheritance is the key feature of Cultural Algorithms which is built on the principles of Renfrew's THINKS model [11] and which allows for a two-way system of learning and adaptation to take place. Reynolds and his team observed that, various knowledge sources like topographic knowledge, situational knowledge, and the fine-grained knowledge interact at the cultural level representing the Cultural Swarm, i.e., swarming behavior [12–14]. The early cultural evolution was modeled and studied based on single agent and multiagents to explore the impact of decision-making methods and resource sharing methods on population survival [15, 16].

Ray and Liew [17], inspired by nature's complex intra- and intersociety interactions, proposed an algorithm called society and civilization algorithm (SCA), in which, the artificial societies are build and the leader of the society is identified. The individuals collaborate with the leader and other individuals in the society to evolve. The leader will extend collaboration and communication with the other society leaders, in the civilization in order to improve. This may lead to migration of leaders and individuals to better performing societies. The SCA was implemented on engineering optimizations problems to demonstrate effectiveness of the algorithm.

Nature inspired evolutionary algorithms that emulate the behavior of living organisms and species integrated interactions among the agents for generating the next generation solution space. The outcome of these, interactions are either positive or negative based a random value compared with the algorithm specific control parameter. What if, the outcome of these interactions is indecisive? Indecisive interactions are also one of the outcomes of an interaction which may also need an attention to be accounted in the evolution process. Humans are intelligent species and are capable of withholding the information from the indecisive interaction in the memory and further evaluate, before adopting the change. The indecisive interactions drive the individual to seek a second opinion to build on the exchanged information from the previous interaction to emerge and evolve which is a common behavior with the people in the society.

In a society, in the process of evolution, individuals interact and exchange information. Human's interactions are generally selective and are free to explore randomly based on the individual bias. The individuals initially extend the interactions within the neighborhood because of affinity and trust worthiness. But, the interactions, does not get limited to neighborhood only. The individual is free to extend the interactions with the society in the process of evolution, particularly, when these interactions does not offer better prospects for growth or purely based on personal choice.

Motivated by such human characteristics, this paper introduces social evolution (SE) algorithm that employs the concept of second opinion and freedom of interactions to enable the individual's evolution leading to social evolution.

2 Social Evolution Algorithm

The pseudo code of SE algorithm is presented below:

1. Initialization:
   - Initialize Control parameters
     Maximum Cycle Number (MCN)
     Neighborhood Cooperation factor (NCR)
     Quality of Interaction (QI)
     Indecisive Factor (IFD)
     Second Opinion (Neighbor Best)–NB
     Second Opinion (Average Best)–AB
     Second Opinion (Society Best)–SB
   - Initialize the population
2. Evaluation Phase:
   - Evaluate the fitness of each individual
   - Calculate the probability of each individual and average solution
   - Store the best solution in the community
3. Cycle = 0
4. REPEAT when Cycle < Maximum cycle number

5. Interaction Phase:
   REPEAT (for all the individuals in the society)

   - Allow Individuals to interact based on their ability to interact (probabilistically).
   - Freedom of interaction:
     - Identify the neighbors based on von Neumann architecture
     - Store the best in the neighborhood
     - Choose between the random neighbor and a random individual based on the cooperation factor and fitness
   - Produce the new solution v_i for each individual using (1)

$$V_{ij} = X_{ij} + \emptyset_{ij}(X_{ij} - X_{kj}) \quad if \quad R_j < \text{ QI}, \qquad \textit{Otherwise}, \text{ evaluate IDF} \tag{1}$$

     [$\emptyset_{ij}$– is a Emotion Quotient—a random number in the range [−1, 1]. $k \in$ {1, 2,... SN} (SN: Number of individuals in a society) is randomly chosen index within the neighborhood/society of the individual. Although k is determined randomly, it has to be different from i. R_j is a randomly chosen real number in the range [0, 1] and $j \in$ {1, 2,... D}(D: Number of dimensions in a problem). [QI, Quality of interaction, is a control parameter].
   - Process of second opinion
     - Evaluate the Indecisive Factor (IDF)
       If Rand( ) > IDF, – $V_{ij} = X_{ij}$
       If Rand( ) < IDF – Look out for second opinion
     - Seek Second opinion and build on the previous interaction
       · If Rand( ) < NB

$$V_{ij} = V_{ij} + \emptyset_{ij}(V_{ij} - X_{ij}) \tag{2}$$

         1 ∈ {1, 2, ...SN} is index of the neighborhood best
       · If Rand( ) < AB

$$V_{ij} = V_{ij} + \emptyset_{ij}(V_{ij} - A_j) \tag{3}$$

         A is the average individual and $j \in$ {1, 2,... D}
       · If Rand( ) < SB

$$V_{ij} = V_{ij} + \emptyset_{ij}(V_{ij} - X_{mj}) \tag{4}$$

         $m \in$ {1, 2, ...SN} is index of the best in the society.
   - Evaluate the fitness of the individual before and after interaction and consider the best for next generation

6. UNTIL for all the individuals in the society are processed
7. Calculate the probability if each individual and average solution
8. Store the best solution in the society
9. Cycle = Cycle + 1

10. UNTIL (Maximum cycle number—The termination criteria is satisfied).

In the initialization phase, the basic control parameters such as number of maximum cycle number, quality of interaction, indecisive factor, second opinion ranges are initialized and the random population of individuals are generated. In the evaluation phase, each individual's fitness and its probability is calculated along with the average fitness of the individuals.

In the interaction phase, individuals interact with the neighbors probabilistically. In this algorithm, von Neumann neighborhood architecture is adopted for building the neighborhood. The individual first identifies the neighborhood individuals and randomly identifies a neighbor to interact. Before interaction, the individual evaluates the neighbor based on the cooperation factor and the ability or productivity of the neighbor. Based on the analysis or simply based on individual's bias, individual may interact with the identified neighbor or may consider a random individual in the society for an interaction. The individual interaction operator is inspired by the artificial bee colony optimization algorithm's employee/onlooker bee operator [10] as this operator exhibits the human interactions model. In the interaction operator, unlike the bee agents in ABC, the individual will not interact with any random solution in the society instead, they may interact more with the random neighbor in the von Neumann neighborhood architecture because of affinity and trust worthiness, but they are free to explore the society based on NCF. Also, once the individual is selected for the interaction, the individual solution interacts with the selected individual for all the dimensions of the problem unlike the operator used in ABC algorithm [10]. Once the interaction is performed, individual evaluate the quality of interaction (QI). If the quality of interaction is inferior, interaction's indecisive factor IDF is evaluated to decide on the interaction as negative or indecisive. All the indecisive interactions will undergo a second opinion process.

In the second opinion process, the individual can consult an expert either from the neighborhood or from the society or a non-existing individual with the average capabilities to further evaluate the indecisive interaction before adopting the change to emerge and evolve. After the interaction phase, evaluate the fitness of the updated solutions and compare with the respective original solution to consider the best for next generation. Before the above process is repeated until a termination condition (maximum cycle number), calculate the probabilities of the individuals, average solution the best in the society for the next generation.

The proposed SE algorithm is applied on four numerical benchmark problems and three to test the effectiveness and adoptability of the algorithm.

Table 1 Experimental parameters

Parameter	values
Population	100
Maximum cycle number (MCN)	10,000
Neighborhood cooperation factor (NCR)	0.75
Quality if interaction	0.8
Indecisive factor	0.5
Second opinion (Neighbor best)–NB	< 0.3
Second opinion (Average best)–AB	< 0.5
Second opinion (Best)–SB	< 1.0

3 Experiments

3.1 Unconstrained Benchmark Optimization Problems

Initially, the performance of the algorithm is tested on four standard numerical benchmark problems given in Table 2 using the experimental parameters presented in Table 1. Sphere is a convex, separable, unimodal function which has no local minimum except the global one. Schwefel is a multi model, non-separable function whose surface is composed of a great number of peaks and valleys. For this problem, many search algorithms get trapped in to the second best minimum far from the global minimum. Rastrigin is a multimode separable function, which was constructed from Sphere adding a modulator term. Its contour has a large number of local minima whose value increases with the distance to the global minimum. Dixon-Price is multimode, non-separable, and non-symmetric function.

Table 2 Benchmark functions

Function name	Interval	Function
Sphere	$[-100, 100]^n$	$\text{Min F} = \sum_{i=1}^{n} x_i^2$
Schwefel	$[-500, 500]^n$	$\text{Min F} = \sum_{i=1}^{n} \{-x_i \sin(\sqrt{\lvert x_i \rvert}\}$
Dixon–price	$[-10, 10]^n$	$\text{Min F} = (x_1 - 1)^2 + \sum_{i=2}^{n} i(2x_i^2 - x_{i-1})^2$
Rastrigin	$[-5.12, 5.12]^n$	$\text{Min F} = \sum_{i=1}^{n} \left\{\left(x_i^2 - 10\cos(2\pi x_i) + 10\right)\right\}$

Table 3 Results (Mean of the best values) of PSO, DE and ABC and SE algorithms on unconstrained numerical benchmark problems

Function name	PSO [18]	DE [18]	ABC [18]	SE
Sphere	0	0	0	0
Schwefel	−2654.030	−4177.990	−4189.830	−4189.830
Dixon–price	0.666	0.666	0	4.67E-01
Rastrigin	7.363	0	0	0

The social evolution algorithm executed 30 independent runs, to ascertain the performance of the algorithm on each of the listed problems with 10 dimensions. For easy comparisons, the experimental parameters population size and the maximum number of cycles are defined as per the parameters defined for DE, PSO, and ABC algorithms [18]. The mean of the best values for each of the problems is reported in Table 3 and compared against some of the evolutionary algorithms such as, differential evolution (DE), particle swarm optimization (PSO) and artificial bee colony optimization (ABC). The reported results suggest that, the algorithm finds the global optimum values for the three functions Sphere, Schwefel, and Rastrigin successfully. When compared with individual algorithms, the ABC algorithm performed better than SE on one function and SE performed better than PSO on three functions and better than DE on two functions.

4 Conclusions

This paper proposes a novel social evolution (SE) algorithm that mimics the human interactions, behavior, and their biases. This algorithm adopts two basic human characteristics. First, individual's ability and bias to evaluate the neighbor before establishing the interaction to evolve. Second, the ability that discriminate the quality of interaction and identifying indecisive interactions and further seeking a second opinion from an expert before adopting the change to emerge and evolve. The proposed algorithm mimics such characteristics of humans tested on four numerical test functions and compared the results with the state-of- the-art nature inspired algorithms. The results indicate that the proposed social evolution algorithm is better than or quite comparable to the existing state-of-the-art algorithms.

Acknowledgments Authors gratefully acknowledge the inspiration and guidance of the Most Revered Prof. P. S. Satsangi, the Chairman, Advisory Committee on Education, Dayalbagh, Agra, India.

References

1. Yoshida, Z.: Nonlinear Science: the Challenge of Complex Systems. Springer, Heidelberg (2010)
2. de Castro L.N.: Fundamentals of natural computing: an overview. Phys. Life Rev. **4**(1), 1–36 (2007)
3. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Kluwer Academic Publishers, Boston, MA (1989)
4. Fogel, D.B.: Evolutionary computation: toward a new philosophy of machine intelligence (3rd edn). IEEE Press, Piscataway, NJ (2006)
5. Beyer, H.-G., Schwefel, H.-P.: Evolution strategies: a comprehensive introduction. J. Nat. Comput. **1**(1), 3–52 (2002)

6. Banzhaf, W., Nordin, P., Keller, R.E., Francone, F.D.: Genetic Programming: An Introduction: On the Automatic Evolution of Computer Programs and Its Applications. Morgan Kaufmann, Heidelberg (1998)
7. Korns, Michael: Abstract Expression Grammar Symbolic Regression, in Genetic Programming Theory and Practice VIII. Springer, New York (2010)
8. White, T., Pagurek, B.: Towards multi-swarm problem solving in networks, In: Proceedings of the 3rd International Conference on Multi-Agent Systems (ICMAS-98), pp. 333–40, (1998)
9. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In Proceedings of 1995 IEEE International Conference Neural Networks IV, pp. 1942–1948, (1995)
10. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J. Global Optim. **39**, 459–471 (2007)
11. Reynolds, R.G.: An introduction to cultural algorithms. In: Proceedings of the Third Annual Conference on Evolutionary Programming, pp. 131–139. San Diego, California (1994)
12. Reynolds, R.G., Peng, B., Brewster, J.J.: Cultural swarms: knowledge-driven problem solving in social systems. IEEE Int. Conf. Syst. Man Cybern. **4**, 3589–3594 (2003)
13. Reynolds, R.G., Peng, B., Brewster, J.: Cultural swarms. Congr. Evol. Comput. **3**, 1965–1971 (2003A)
14. Reynolds, R.G., Jacoban, R., Brewster, J.: Cultural swarms: assessing the impact of culture on social interaction and problem solving. In: Proceedings of the 2003 IEEE Swarm Intelligence, Symposium, pp. 212–219 (2003b)
15. Reynolds, RG, Kobti, Z., Kohler, T.: The effect of culture on the resilience of social systems in the village multi-agent simulation. In: Proceedings of IEEE International Congress on Evolutionary Computation. Portland, OR, vol. 24, pp. 1743–1750, June 19 (2004)
16. Reynolds, R.G., Whallon, R., Mostafa, Z.A., Zadegan, B.M.: Agent-based modeling of early cultural evolution. IEEE Congress on Evolutionary Computation. pp. 1135–1142 (2006)
17. Ray, T., Liew, K.M.: Society and civilization: an optimization algorithm based on the simulation of social behavior. IEEE Trans. Evol. Comput. **7**(4), 386–396 (2003)
18. Akay B., Karaboga D.: Artificial bee colony algorithm for large-scale problems and engineering design optimization. J. Intel. Manuf. pp. 1–14 (2010). DOI: 10.1007/s10845-010-0393-4