



The networked evolutionary algorithm: A network science perspective



Wenbo Du^a, Mingyuan Zhang^a, Wen Ying^a, Matjaž Perc^{a,b,*}, Ke Tang^c,
Xianbin Cao^{a,**}, Dapeng Wu^{d,**}

^aSchool of Electronic and Information Engineering, National Engineering Laboratory for Big Data Application Technologies for Comprehensive Traffic, Beihang University, Beijing 100191, China

^bFaculty of Natural Sciences and Mathematics, University of Maribor, Koroška cesta 160, Maribor SI-2000, Slovenia

^cShenzhen Key Laboratory of Computational Intelligence, Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China

^dDepartment of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611, USA

ARTICLE INFO

Keywords:

Evolutionary algorithm
Network system
Structure
Behavior

ABSTRACT

The evolutionary algorithm is one of the most popular and effective methods to solve complex non-convex optimization problems in different areas of research. In this paper, we systematically explore the evolutionary algorithm as a networked interaction system, where nodes represent information process units and connections denote information transmission links. Within this networked evolutionary algorithm framework, we analyze the effects of structure and information fusion strategies, and further implement it in three typical evolutionary algorithms, namely in the genetic algorithm, the particle swarm optimization algorithm, and in the differential evolution algorithm. Our results demonstrate that the networked evolutionary algorithm framework can significantly improve the performance of these evolutionary algorithms. Our work bridges two traditionally separate areas, evolutionary algorithms and network science, in the hope that it promotes the development of both.

© 2018 Elsevier Inc. All rights reserved.

1. Introduction

Optimization is crucial in scientific research and engineering areas [1–3]. Many real-world problems can be formalized as minimization optimization as follow:

$$\begin{aligned} \min \quad & f(\mathbf{x}), & \mathbf{x} &= [x_1, x_2, \dots, x_D] \\ \text{s.t.} \quad & g_i(\mathbf{x}) \leq 0, & i &= 1, 2, \dots, n, \end{aligned} \quad (1)$$

where D is the dimension of the problem, n is the number of constraints, $f(\mathbf{x})$ is an objective function and $g_i(\mathbf{x})$ are constraint conditions. As many real-world problems are NP hard and grow increasingly complex, the corresponding objective functions are generally non-convex, with non-smooth surfaces and numerous local optima, where most of the gradient-

* Corresponding author at: School of Electronic and Information Engineering, National Engineering Laboratory for Big Data Application Technologies for Comprehensive Traffic, Beihang University, Beijing 100191, China.

** Corresponding authors.

E-mail addresses: matjaz.perc@uni-mb.si (M. Perc), xbcao@buaa.edu.cn (X. Cao), dpwu@ufl.edu (D. Wu).



Fig. 1. Framework of evolutionary algorithms.

based convex optimization methods, such as gradient descent Newton–Raphson method, are incompetent. Hence tremendous efforts have been dedicated to developing more intelligent algorithms to handle multimodal complex optimization problems.

An evolutionary algorithm (EA) is inspired by natural or human intelligence [4]. In an evolutionary algorithm, a solution of the objective function is mapped as an individual, and the value of the objective function corresponds to the evaluation of the individual. Furthermore, a group of solutions are reserved simultaneously, known as individuals in a population. A batch of new solutions is generated in each iteration via crossover and mutation operators, and then some of the existing solutions and new solutions will be reserved by selection mechanisms and get in the next iteration. As the algorithm runs iteratively, the objective function is searched randomly but with a certain trend determined by the algorithm, and the quality of solutions will gradually increase. In the past decades, several EAs have been proposed to improve the performance on optimization problems, such as genetic algorithm [5], particle swarm optimization [6], differential evolution [7], and the artificial bee colony algorithm [8], to name just some (we refer to [9] for a Python-based microframework for building nature-inspired algorithms). These algorithms look very different with various descriptions and mechanisms, yet they share the same framework in form: generating new solutions, evaluating solutions, selecting solutions, as shown in Fig. 1. Essentially, all the EAs devote to make a tradeoff between exploration and exploitation, i.e. how widely to search in the global space and how precisely to dig around the local space.

Information interaction, i.e. communication between individuals, is crucial for EAs. As most of the real-world optimization problem is continuous and slow-varying, a high-quality solution (a position in the search space) that has been found usually indicates a promising region, i.e. it is more probable to find high-quality solutions around this region. From that point of view, the optimization process is similar to the process of gold mining. Suppose that two teams with a same number of miners compete for gold mining. Team A is well organized with a good communication mechanism while team B is a group of disorganized miners. Obviously, team A is more efficient, because team A is able to spread information about where gold has been found, yet in team B, each person could only utilize self-experience.

Furthermore, two aspects are significant to team A's effectiveness of the mining process. The first is structure. The team with fully-connected structure is usually not a good choice, as over-redundant information may interfere one's judgement. Yet an over-sparse structure is inefficient for spreading information. A preferable scheme should be with asymmetric structure where numerous miners could only communicate with several hub miners yet not all miners. The second is information processing strategy. Based on the asymmetric structure above, the structural importance of miners are quite different. Though all miners could follow the same behavior pattern, it would be more efficient if the hub miners become leaders, because hub miners could collect more information thus have better understandings of the mineral distribution.

Similarly, a well-designed EA should also take into account the importance of structural and behavioral heterogeneity. However, these are in fact not well-implemented in most current EAs. Numerous information processing strategies have been proposed to improve the performance, some of which could achieve remarkable results, yet individuals are designed with the same pattern due to the intrinsic homogeneity of these systems [10]. A few works notice the importance of structure, however, most of them still focus on distance-based cluster structure [11,12], multiswarm technique [13,14] or regular structures [15,16]. Hence most EA variants are essentially limited in balancing exploration and exploitation.

In this paper, we consider the EA as a networked information transmission system (NITS). To improve the performance, we employ heterogeneous network structure and design appropriate information fusion strategies of nodes (individuals). Structurally important nodes, i.e. hub nodes, mainly play the role of processing and distributing information. Non-hub nodes provide their information to the hubs while not always follow the hubs' guide, but instead do explorations themselves to find other promising regions sometimes. Based on this idea, we propose the networked evolutionary algorithm (NEA) framework, exploring the effects of population structure on the performance. In the past decades, the advances in network science have shown that structure plays an important role in the functionalities of a system, such as robustness, spreading, cooperation, synchronization, controllability, and so on. It is fairly expected that the population structure will also impact the

Download English Version:

<https://daneshyari.com/en/article/8900580>

Download Persian Version:

<https://daneshyari.com/article/8900580>

[Daneshyari.com](https://daneshyari.com)