Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Distributed parallel cooperative coevolutionary multi-objective large-scale immune algorithm for deployment of wireless sensor networks

Bin Cao [a,b,c,*], Jianwei Zhao [a,b,c], Po Yang [d,*], Zhihan Lv [e], Xin Liu [a], Xinyuan Kang [a,b,c], Shan Yang [a,b,c], Kai Kang [a], Amjad Anvari-Moghaddam [f]

[a] *Hebei University of Technology, 300401 Tianjin, China*
[b] *Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, 510006 Guangzhou, China*
[c] *Hebei Provincial Key Laboratory of Big Data Calculation, 300401 Tianjin, China*
[d] *Department of Computer Science, Liverpool John Moores University, Liverpool, UK*
[e] *School of Data Science and Software Engineering, Qingdao University, 266071 Qingdao, China*
[f] *Department of Energy Technology, Power Electronic Systems, 9220 Aalborg, Denmark*

HIGHLIGHTS

- The proposed algorithm integrates the idea of immune algorithm.
- Distributed parallelism is constructed through three types of decompositions. The decompositions are: objective, variable and fitness evaluation decompositions.
- The proposed algorithm excels other algorithms in both effectiveness and efficiency.

ARTICLE INFO

Article history:
Received 25 June 2017
Received in revised form 4 September 2017
Accepted 9 October 2017
Available online xxxx

Keywords:
Decision variable analysis (DVA)
Cooperative coevolution (CC)
Immune algorithm (IA)
Large-scale optimization
Message passing interface (MPI)
Wireless sensor networks (WSNs)
3D terrain deployment

ABSTRACT

The use of immune algorithms is generally a time-intensive process—especially for problems with numerous variables. In the present paper, we propose a distributed parallel cooperative coevolutionary multi-objective large-scale immune algorithm parallelized utilizing the message passing interface (MPI). The proposed algorithm comprises three layers: objective, group and individual layers. First, to tackle each objective in a multi-objective problem, a subpopulation is used for optimization, and an archive population is used to optimize all the objectives simultaneously. Second, the numerous variables are divided into several groups. Finally, individual evaluations are allocated across many core processing units, and calculations are performed in parallel. Consequently, the computation time is greatly reduced. The proposed algorithm integrates the idea of immune algorithms, exploring sparse areas in the objective space, and uses simulated binary crossover for mutation. The proposed algorithm is employed to optimize the 3D terrain deployment of a wireless sensor network, which is a self-organization network. In our experiments, through comparisons with several state-of-the-art multi-objective evolutionary algorithms—the cooperative coevolutionary generalized differential evolution 3, the cooperative multi-objective differential evolution, the multi-objective evolutionary algorithm based on decision variable analyses and the nondominated sorting genetic algorithm III—the proposed algorithm addresses the deployment optimization problem efficiently and effectively.

1. Introduction

In the wireless sensor network (WSN) deployment optimization procedure [1], wireless sensor nodes can be optimized via self-organization [2] to maximize the *Coverage*, optimize the *Connectivity Uniformity* and minimize the *Deployment Cost*. With the rapid development of sensor and wireless communication technologies,

[*] Corresponding authors.

E-mail addresses: caobin@scse.hebut.edu.cn (B. Cao), poyangcn@gmail.com (P. Yang).

https://doi.org/10.1016/j.future.2017.10.015
0167-739X/© 2017 Published by Elsevier B.V.

WSNs have been applied to various fields. The work of [3] presented an air temperature monitoring application for WSNs. Shen et al. [4] described the wireless sensor nodes for a medical service. Zhang et al. [5] illustrated the WSN k-barrier coverage problem. Zhou et al. [6] researched the energy issue, regarding which clustering and data compression were studied. Zhang et al. [7] utilized mobile sinks to alleviate the communication burden.

In addition, the response of the human immune system to antigens can be viewed as a process of self-organization. Based on this concept, the clonal selection algorithm (CLONALG) [8], which can be used for global optimization problems (GOPs) and multi-objective optimization problems (MOPs) [9], was proposed. Other nature-inspired algorithms also follow the self-organizing procedure. For example, Xue et al. [10] described the self-adaptive artificial bee colony algorithm, which is different from the immune algorithm.

In the real world, many problems require several (usually conflicting) objectives to be considered simultaneously. Multi-objective evolutionary algorithms (MOEAs) [11–13] are capable of producing a plurality of solutions during one run, which is convenient for approximating the Pareto front (PF). For NP-hard problems, evolutionary algorithms (EAs) [14–17] can usually converge to near-optimal solutions using limited computational resources [18] within a reasonable time compared to brute force and deterministic methods.

The first multi-objective immune algorithm (MOIA) was proposed in [19]. In this study, the immune algorithm (IA) was integrated into the genetic algorithm (GA) to improve the selection of individuals for evolution. Gong et al. [20] presented the nondominated neighbor immune algorithm (NNIA), selecting a small quantity of nondominated individuals in a sparse area for cloning, recombination and mutation. In [21], simulated binary crossover (SBX) and differential evolution (DE) were combined and applied to cloned individuals in a hybrid evolutionary framework for MOIAs (HEIA), which performed well for both unimodal and multimodal problems.

EAs are based on an iterative evolution of the population (the solutions), which is time-consuming—especially for expensive problems. Distributed evolutionary algorithms (dEAs) [22,23] allocate the tedious computational burden across numerous computational nodes, greatly reducing the required time. Cloudde [24] used DEs with various parameters to optimize multiple populations in a distributed parallel manner, yielding a promising performance from both the effectiveness and efficiency aspects. [25] provided a comprehensive study concerning parallel/distributed MOEAs. Utilizing the multi-objective optimization algorithm based on decomposition (MOEA/D) [13], parallel MOEA/Ds (pMOEA/Ds) [26,27] were proposed.

With the arrival of "big data", many complex problems have emerged; solving such problems is both time-consuming and storage-consuming [28,29]. Similarly, many MOPs now have numerous variables (e.g., more than 100 variables [30]). Some examples include classification [31], clustering [32], and recommendation systems [33]. However, the goal of traditional MOEAs is to solve multi-objective small-scale optimization problems (MOSSOPs). Consequently, the traditional algorithms may be incapable of tackling multi-objective large-scale optimization problems (MOLSOPs) because of the "curse of dimensionality". To optimize numerous variables, some promising approaches first separate the variables into groups and then optimize them in a cooperative coevolutionary (CC) [34] manner. For large-scale global optimization problems (LSGOPs), many grouping mechanisms have been applied, including fixed grouping [34], random grouping [35], the Delta method [36], dynamic grouping [37], differential grouping (DG) [38], global differential grouping (GDG) [39] and graph-based differential grouping (gDG) [40]. Antonio et al. proposed the cooperative coevolutionary generalized differential evolution 3 (CCGDE3) method [41], which used fixed grouping.

MOLSOPs differ from LSGOPs in that no single solution optimizes all the conflicting objectives; instead, a solution set should be generated to approximate the PF. In MOLSOPs, variables have different properties [42], which can be classified as follows:

1. position variables, which affect only the diversity of the solution set;
2. distance variables, which affect only the convergence of the solution set; and
3. mixed variables, which affect both the diversity and the convergence of the solution set.

Therefore, position variables should be permuted to approximate the PF as comprehensively as possible. However, distance variables should be optimized so that they can closely approach the PF.

To identify these variable types, the multi-objective evolutionary algorithm based on decision variable analyses (MOEA/DVA) [30] utilizes a mechanism called decision variable analyses (DVA). The position as well as mixed variables are categorized as diversity-related variables, while distance variables, as convergence-related variables. The convergence-related variables are allocated to multiple groups that are then optimized under the CC framework.

The use of multiple populations can impact the optimization performance. In cooperative multi-objective differential evolution (CMODE) [43], each objective is optimized by a subpopulation, and an archive is used to maintain good solutions and optimize all objectives. This approach has yielded good experimental results.

Compared to MOSSOPs, designing parallel/distributed MOEAs for MOLSOPs will be more beneficial. In this paper, we propose the distributed parallel cooperative coevolutionary multi-objective large-scale immune algorithm (DPCCMOLSIA), which is aimed at solving MOLSOPs effectively and efficiently.

The contributions of this paper can be highlighted as follows:

1. Each objective is optimized by a subpopulation. Thus, the exploration with respect to each objective is enhanced, and all objectives are comprehensively optimized by an archive. Variables are grouped according to their properties and interactions, contributing to effective optimization.
2. The idea of the IA is introduced, more computational resources are used to explore sparse areas in the objective space, and *SBX* is utilized for evolution.
3. We construct a three-layer parallel structure. The evaluations of individuals in different groups of multiple populations can then be performed in parallel, which greatly reduces the computation time.

The remainder of this paper is organized as follows: Section 2 provides some preliminary information required for this paper. The details of the DPCCMOLSIA are discussed in Section 3. Then, in Section 4, we describe the experimental study and present the corresponding analyses. Finally, Section 5 concludes this paper.

2. Preliminaries

2.1. MOP and variable properties

An MOP involves several objectives that usually conflict with each other. Therefore, addressing an MOP comprises obtaining a solution set that approximates the PF. For the minimization problem, we have the following formula:

$$\text{Minimize } F(\boldsymbol{X}) = \{f_1(\boldsymbol{X}), f_2(\boldsymbol{X}), \ldots, f_M(\boldsymbol{X})\}, \tag{1}$$