A New Physarum Network Based Genetic Algorithm for Bandwidth-Delay Constrained Least-Cost Multicast Routing

Mingxin Liang[1], Chao Gao[1], Yuxin Liu[1], Li Tao[1], and Zili Zhang[1,2(✉)]

[1] School of Computer and Information Science, Southwest University, Chongqing 400715, China
zhangzl@swu.edu.cn
[2] School of Information Technology, Deakin University, Geelong, VIC 3217, Australia

Abstract. Bandwidth-delay constrained least-cost multicast routing is a typical NP-complete problem. Although some swarm-based intelligent algorithms (e.g., genetic algorithm (GA)) are proposed to solve this problem, the shortcomings of local search affect the computational effectiveness. Taking the ability of building a robust network of *Physarum* network model (PN), a new hybrid algorithm, *Physarum* network-based genetic algorithm (named as PNGA), is proposed in this paper. In PNGA, an updating strategy based on PN is used for improving the crossover operator of traditional GA, in which the same parts of parent chromosomes are reserved and the new offspring by the *Physarum* network model is generated. In order to estimate the effectiveness of our proposed optimized strategy, some typical genetic algorithms and the proposed PNGA are compared for solving multicast routing. The experiments show that PNGA has more efficient than original GA. More importantly, the PNGA is more robustness that is very important for solving the multicast routing problem.

Keywords: Genetic algorithm · *Physarum* network model · Multicast routing

1 Introduction

Multicasting is one type of services in MANETs, which are very popular due to the no-restricted mobility and feasible deployment. With the growing of distributed multimedia application, the efficient and effective support of QoS (i.e., Quality of Service) has became more and more crucial for MANETs. The key issue in the design of network architectures of MANETs is how to manage the resources efficiently in order to meet the requirements of QoS during each connection. In general, to deliver the same data stream to different destinations efficiently, a tree structure is used for multicasting. More importantly, some constrains are often added to the entire tree for multicasting in order to meet

© Springer International Publishing Switzerland 2015
Y. Tan et al. (Eds.): ICSI-CCI 2015, Part II, LNCS 9141, pp. 273–280, 2015.
DOI: 10.1007/978-3-319-20472-7_29

the requirements of QoS. Thus, the object of multicast routing problem is to compute a tree structure (named as the multicast tree) under the conditions of the minimum communication resources and QoS requirements of a network [2], which is a typical NP-complete problem [3].

In the field of artificial intelligence, the genetic algorithm is a powerful tool for solving NP-complete problems. However, the shortcomings of local search affect the computational effectiveness. Recently, more and more scientists focus on the self-organization capability of a species of plasmodium, which is a 'vegetative' phase of *Physarum*. This plasmodium shows an amazing intelligence in the process of building a robust protoplasmic network for connecting food sources in order to deliver nutrients to all its body [4].

In this paper, we design an updated crossover operator, based on *Physarum* Network (PN) [5], for overcoming the shortcomings of traditional GA. Using this method, we incorporate PN into GAs for solving the multicast routing problem. Some experiments show that the hybrid algorithm has a stronger ability to exploit the optimal solution effectively.

The organization of this paper is as follows. Section 2 introduces the formulation and measurements of multicast routing problem. Section 3 formulates the hybrid algorithm. Section 4 provides some experiments to estimate the effectiveness of hybrid algorithm. Section 5 concludes this paper.

2 Problem Statement

In general, a QoS multicast routing problem involves in several constrains, such as delay jitter, packet loss, bandwidth, and cost. In this study, we simplify QoS constrains and present a feasible QoS multicast routing model. According to [1], we focus on three most important factors: cost, bandwidth and delay. As the cost is the most important metric for the effectiveness of a network, our research focuses on the bandwidth-delay constrained least-cost multicast routing problem.

A network is usually represented as a graph $G = (V, E)$, where $V = \{v_1, v_2, ..., v_n\}$ denotes a set of nodes representing routers or switches and $E = \{e_{ij} = (v_i, v_j) | v_i, v_j \in V, i \neq j\}$ denotes a set of edges representing physical or logical connectivity between nodes. Let a node $s \in V$ be the source and a set $DE \subseteq V - \{s\}$ be the set of multicast destinations. A multicast tree, denoted as $T(s, DE)$, is a sub-graph of G connecting a node s to each node in DE. The path from s to any destination node $d \in DE$ is denoted as $p_T(s, d)$. And the object of multicast routing problem is to find a $T(s, DE)$ with a minimum cost, which has a set of paths with acceptable bandwidth and delay from a node s to each node in DE.

The delay of a path from a node s to any destination node d in DE, denoted as $delay(p_T(s, d))$, is simply defined as the sum of delays in the $p_T(s, d)$, i.e., $delay(p_T(s, d)) = \sum_{e \in p_T(s,d)} delay(e)$. Meanwhile the bandwidth of a path from a node s to any destination node d in DE, denoted as $bandwidth(p_T(s, d))$, is defined as the minimum of bandwidth along $p_T(s, d)$, i.e., $bandwidth(p_T(s, d)) = \min\{bandwidth(e) | e \in p_T(s, d)\}$. And, according to existing studies in [6,7], we unify the cost of a multicast tree as the sum of costs in the tree, i.e., $cost(T) = \sum_{e \in T} cost(e)$, for comparing effects of different GAs.

Let Δ_d be the upper limit of delay constraint and Δ_b be the lower limit of bandwidth constraint for each path. And the bandwidth-delay constrained least-cost multicast routing problem is defined as (1). We want to minimize the cost under the condition of satisfying bandwidth-delay constraints.

$$\begin{array}{l} \quad \min \cos t(T) \\ st. \\ \quad \begin{cases} delay(p_T(s,d)) \le \Delta_d \qquad for\ \forall\ d \in DE \\ bandwidth(p_T(s,d)) \ge \Delta_b\ for\ \forall\ d \in DE \end{cases} \end{array} \tag{1}$$

3 Formulation of PNGA

This section introduces the basic idea of PNGA from two aspects: original PN model and PNGAs. In detail, Sect. 3.1 presents the original PN model. And, Sect. 3.2 shows how to improve GAs by PN.

3.1 *Physarum* Network Model

The *Physarum* network model is inspired by the maze-solving experiment [4]. Tero et al. capture the positive feedback mechanism of *Physarum* in foraging and build the PN model [5]. In addition, The model, designed for solving maze problem, can be used for building a multicast tree in our study. The details of PM are described as follows.

In PM, Q_{ij} represents the flux of pipeline, connecting nodes i and j, and D_{ij} stands for the conductivity of the pipeline. Moreover a node s and a set DE present the inlet and outlets of pipelines respectively. According to the Kirchhoff's law, the flux of input at node s is equal to the total flux of output at DE and, at any other nodes, the sum of flowing into that node is equal to the sum of flowing out of that node. This process can be denoted as (2) where N stands for the cardinality of DE.

$$\sum_i Q_{ij} = \begin{cases} +1 & for\ j == s \\ \frac{-1}{N-1} & for\ j \in DE \\ 0 & for\ \ others \end{cases} \tag{2}$$

In each iteration step, Q_{ij} and p_i can be calculated according to Poiseuille's law based on (2) and (3), where L_{ij} represents the length of pipeline contacting nodes i and j, and p_i represents the pressure of node i. As the iteration going on, the conductivities of pipelines adapt to the flux based on (4). Then, the conductivities will feed back to the flux based on (3) at the next iteration step.

$$Q_{ij} = (\frac{D_{ij}}{L_{ij}} + \frac{D_{ji}}{L_{ji}})(p_i - p_j) \tag{3}$$

$$\frac{dD_{ij}}{dt} = |Q_{ij}| - D_{ij} \tag{4}$$

After above processes, one iterative step is completed. This process will continue loop iteration until the terminal condition is satisfied. In this study, the terminal condition is $|D^{t}{}_{ij} - D^{t+1}{}_{ij}| < 10^{-6}$ for any i and j, where $D^{t+1}{}_{ij}$ stands for D_{ij} at iteration step $t+1$. After the loop iteration, critical pipelines will be reserved, and others will disappear. Finally, we can obtain a *Physarum* spanning tree. The description of PN model for a shortest path tree is shown in Alg. 1.

Algorithm 1. *Physarum* network model

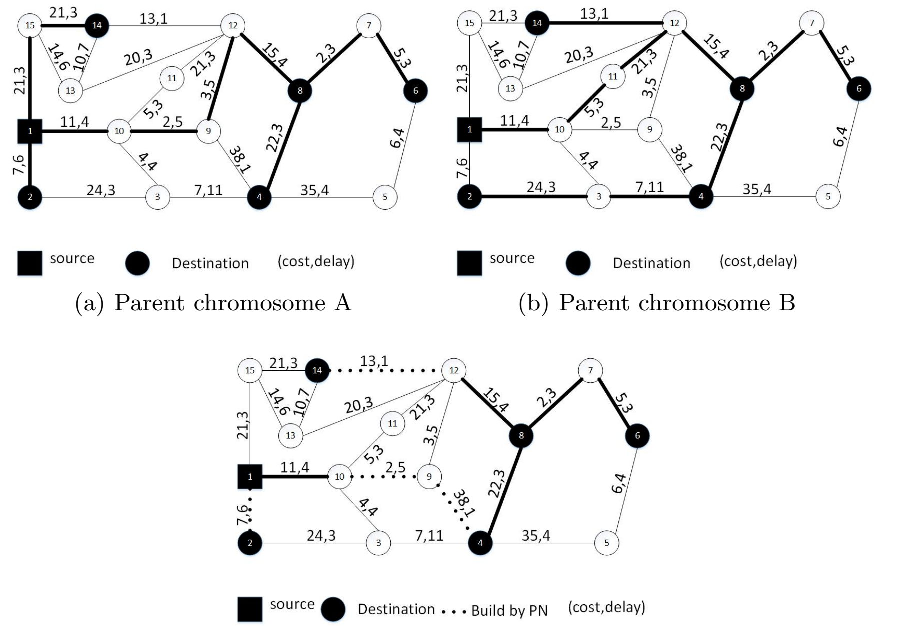
Input: A graph NG, a source node s, a set of destination nodes DE.
Output: A shortest path tree connecting a source node s to each node in DE.
Step 1: Initializing with $D_{ij} = (0, 1]$, $Q_{ij} = 0$, $p_i = 0$.
Step 2: Computing Q_{ij} and p_i based on (2) and (4).
Step 3: **If** $Q_{ij} < 0$, **then** $Q_{ij} = 0$.
Step 4: Updating D_{ij} based on (4).
Step 5: **If** the terminal condition is not satisfied, **then** going to **step 2**.
Step 6: Outputting the shortest path tree.

3.2 PNGA

The GA is a powerful tool for solving PN-complete problems and a kind of searching algorithm that employs the ideas of natural selection and the genetic operators of crossover and mutation. Taking advantages of PN model and GAs, we propose a universal strategy for crossover operator in GAs. The new crossover operator is named as PNcrossover. And the novel hybrid algorithms with PNcrossover (denoted as PNGAs) are used to solve the multicast routing problem. The main ideal of PNcrossover is to reserve the same links between parent chromosomes and to integrate the offspring through PN model based on the reserved links. Other parts of PNGAs are same as the original ones. An example of crossover process is shown in Fig. 1. The details of PNcrossover are described as follows.

Firstly, for applying PN model, we need a new graph, denoted as NG, with the same topological structure as the network graph in the multicast routing problem. Because GAs may generate some unadaptable chromosomes, which do not satisfy with the constraints, there are two strategies to fit for different GAs. For the GAs, which do not generate unadaptable chromosomes, such as GAMRA [6], L_{ij} in NG is set equal to the delay of e_{ij}. For others, such as EEGA [7], L_{ij} in NG is set equal to the product of cost and delay of e_{ij}.

And then, PNcrossover selects the same links of parent chromosomes and reserves them in the offspring chromosomes. Since these same links may be in some separated sub-trees, PN model is used to transform these sub-trees into a multicast tree. In order to reserve the same links in PN model, the length of reserved links is set equal to zero in NG. Substituting the graph NG, source



(a) Parent chromosome A

(b) Parent chromosome B

(c) A new offspring generated by parent chromosomes A and B through PNcrossover

Fig. 1. An example of crossover operation

and destinations into PN model, a complete multicast tree will be constructed. Alg. 2 describes detailed steps of crossover operator in PNGAs.

4 Simulation Experiments

4.1 Datasets

In order to estimate the effectiveness of PNcrossover scheme, we integrate PNcrossover into two different GAs [6,7] and implement them on two datasets. The first (denoted as $D1$) is a random graph with 20 nodes, which is constructed based on [6]. In $D1$, costs and delays of links are uniformly distributed between 0.3 and 1. The second (denoted as $D2$) is shown in Fig. 1, with $\varDelta_d$ equaling to 24 [7]. All experiments are under the same environment, i.e., all parameters of PNGAs are same as these of original GAs. And all results in our experiments are averaged over 50 times.

4.2 Experiments Analysis

Figure 2 shows the minimum ($S_{\min}$), average ($S_{average}$) and variance (S_{variance}) of the results calculated by PN-GAMRA and original GAMRA [6]. Although

Algorithm 2. PNcrossover

Input: A network graph G, a source node s, a set of destination nodes DE, parent chromosomes T_a and T_b.
Output:An Offspring chromosome T_c.
Step 1: Creating a graph NG with the same topological structure as the network graph G in the multicast routing problem.
Step 2: **If** the original crossover operator do not generate unadaptable chromosomes, **then** let L_{ij} equal to the delay of e_{ij}.
else let L_{ij} in NG equal to the product of cost and delay of e_{ij}.
Step 3: Let the length of same links between T_a and T_b equal to zero.
Step 4: Substituting the NG, s and DE nodes into PN model.
Step 5: Outputting a new multicast tree, T_c, based on PN model.

these two GAs can find approximate optimal solutions, $S_{\min}$ and $S_{average}$ of PN-GAMRA are less than that of original GAMRA. That means PN-GAMRA has a stronger ability to exploit the optimal solution. Moreover, $S_{\mathrm{var}iance}$ of PN-GAMRA is less than that of GAMRA, which shows that the PN-GAMRA is more robust than its original algorithm.

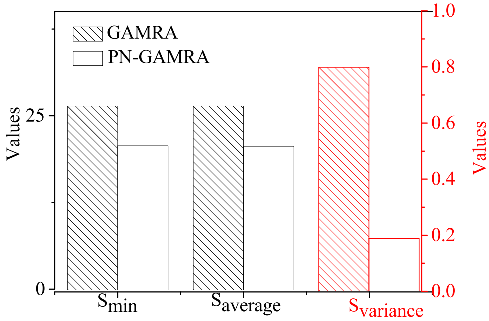
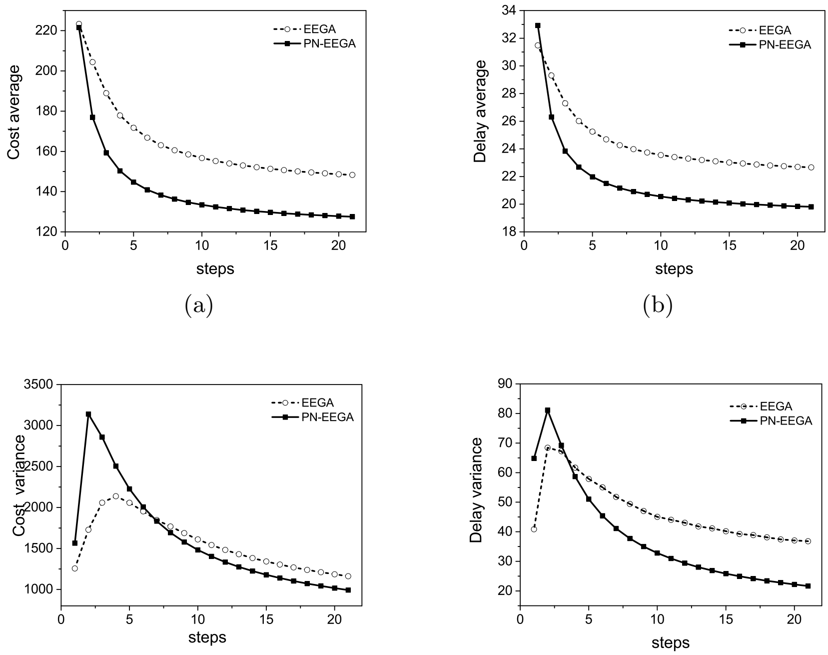


Fig. 2. Comparing results of PN-GAMRA and GAMRA on $D1$

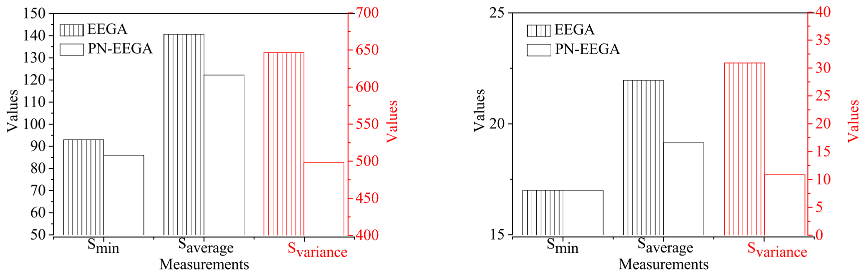
As EEGA [7] may generate unadaptable solutions in the evolution, we compare the costs and delays of solutions. In order to further verify the accuracy and robustness of PNGA, Fig. 3 plots the convergent process of averages and variances with the increment of iterative steps. As shown in Fig. 3, average and variance of PN-EEGA decrease more obviously than that of EEGA. In detail, in the earlier iteration, there are slight difference between averages of PN-EEGA and EEGA. With the iterative steps going on, the average of PN-EEGA are less than that of EEGA obviously. PN-EEGA exhibits a better accuracy. Furthermore, the variance of PN-EEGA is also less than that of EEGA, which shows that PN-EEGA has more stronger robust. Moreover, Figure 4(a) and Fig. 4(b)



(a) (b)

(c) (d)

Fig. 3. Delays and costs calculated by EEGA and PN-EEGA on $D2$



(a) Cost comparison (b) Delay comparison

Fig. 4. Comparing results of EEGA and PN-EEGA on $D2$

plot $S_{\min}$, $S_{average}$ and $S_{\mathrm{var}iance}$ of PN-EEGA are less than that of EEGA in both costs and delays. These results show that the PNcrossover scheme can strengthen the searching ability for finding the optimal solution and improve the robustness of original GA.

5 Conclusion

Inspired by *Physarum* polycephalum forming optimized network in foraging food sources, a new *Physarum* Network based genetic algorithm was proposed to solve multicast routing NP-hard problem. In the proposed PNGA, a new crossover operator was introduced, which can improve the effectiveness of GA. This was verified through experiments on two datasets. As PNGAs need more time on calculating the PN model, more improvements will be implemented to reduce the computational cost in future.

Acknowledgments. This workThis work was supported by the National High Technology Research and Development Program of China (No. 2013AA013801), National Science and Technology Support Program (No. 2012BAD35B08), National Natural Science Foundation of China (Nos. 61402379, 61403315), and Natural Science Foundation Project of CQ CSTC (Nos. cstc2012jjA40013, cstc2013jcyjA40022).

References

1. Wang, Z.Y., Shi, B.X., Zhao, E.: Bandwidth-delay-constrained least-cost multicast routing based on heuristic genetic algorithm. Computer Communications **24**(7), 685–692 (2001)
2. Peng, B., Li, L.: Combination of genetic algorithm and ant colony optimization for QoS multicast routing. In: Cho, Y.I., Matson, E. (eds.) Soft Computing in Artificial Intelligence. AISC, vol. 270, pp. 49–56. Springer, Heidelberg (2014)
3. Wang, Z., Crowcroft, J.: Quality-of-service routing for supporting multimedia applications. IEEE Journal on Selected Areas in Communications **14**(7), 1228–1234 (1996)
4. Nakagaki, T., Yamada, H., Toth, A.: Intelligence: Maze-solving by an amoeboid organism. Nature **407**(6803), 470–470 (2000)
5. Tero, A., Kobayashi, R., Nakagaki, T.: A mathematical model for adaptive transport network in path finding by true slime mold. Journal of Theoretical Biology **244**(4), 553–564 (2007)
6. Hwang, R.H., Do, W.Y., Yang, S.C.: Multicast routing based on genetic algorithms. Journal of Information Science and Engineering **16**(6), 885–901 (2000)
7. Lu, T., Zhu, J.: Genetic algorithm for energy-efficient QoS multicast routing. Communications Letters **17**(1), 31–34 (2013)
8. Salama, H.F.: Multicast routing for real-time communication of high-speed networks. Ph D Thesis. North Carolina State University (1996)
9. Qian, T., Zhang, Z., Gao, C., Wu, Y., Liu, Y.: An ant colony system based on the *physarum* network. In: Tan, Y., Shi, Y., Mo, H. (eds.) ICSI 2013, Part I. LNCS, vol. 7928, pp. 297–305. Springer, Heidelberg (2013)
10. Zhang, Z., Gao, C., Liu, Y., Qian, T.: A universal optimization strategy for ant colony optimization algorithms based on the Physarum-inspired mathematical model. Bioinspiration & Biomimetics **9**(3), 036006 (2014)
11. Karthikeyan, P., Baskar, S.: Genetic algorithm with ensemble of immigrant strategies for multicast routing in Ad hoc networks. Soft Computing **19**(2), 489–498 (2015)
12. Liu, Y., Zhang, Z., Gao, C., Wu, Y., Qian, T.: A *physarum* network evolution model based on IBTM. In: Tan, Y., Shi, Y., Mo, H. (eds.) ICSI 2013, Part II. LNCS, vol. 7929, pp. 19–26. Springer, Heidelberg (2013)