Advances in Computer Science and Engineering
Volume 9, Number 1, 2012, Pages 67-82
Published Online: November 2012
Available online at http://pphmj.com/journals/acse.htm
Published by Pushpa Publishing House, Allahabad, INDIA

ENERGY-EFFICIENT GENETIC ALGORITHM FOR QoS MULTICAST ROUTING

Ting Lu* **and Jie Zhu**

Department of Electronic Engineering
Shanghai Jiao Tong University
Shanghai 200240, P. R. China
e-mail: ting.lu31@gmail.com

Abstract

Since the lack of considering energy consumption in wireless ad hoc networks makes it easy to exhaust battery energy and result in partitioning of the entire network, power-aware multicasting is proposed to reduce power consumption. In this paper, we study the quality of service (QoS) multicast routing in ad hoc networks. It is an NP-complete problem. We present an energy-efficient genetic algorithm mechanism to resolve these problems. The proposed genetic algorithm depends on bounded end-to-end delay and minimum energy cost of the multicast tree. Simulation results show that the proposed algorithm is effective and efficient.

1. Introduction

A mobile ad hoc network (MANET) [1] is a self-configuring network of mobile nodes, which can form a dynamic topology. All the nodes cooperatively maintain network connectively without the aid of any fixed infrastructure units such as base stations or access points in advance. Each

Keywords and phrases: mobile ad hoc networks, multicast tree, quality of service, genetic algorithm, energy cost.

*Corresponding author

Received July 1, 2012

node has a routing function whereby it communicates by forwarding packets via intermediate nodes. If two nodes are within the transmission range of each other, then they communicate directly. Otherwise, other nodes are needed to forward their packets. MANET is characterized by non-restricted mobility and easy deployment, which makes them very promising.

Power awareness is crucial in a mobile wireless network, particularly in a MANET. Nodes need to reduce their power consumption to prolong their battery lifetime [2]. Therefore, the transmission power should be carefully chosen since the large transmission power level leads to the waste of battery energy. The energy-efficient multicasting tree problem was presented in [3]. Several heuristic algorithms for constructing source-based energy-efficient multicast trees have been developed [4-7]. Since most multimedia applications are delay-sensitive, end-to-end delay should be considered in multicast routing to provide better QoS. However, energy-efficient multicast routing has not always considered the delay metric. Furthermore, the design of quality of service (QoS) multicast routing with multi-constrained metrics, i.e., delay-constrained least-cost multicast routing [8], bandwidth-delay-constrained least-cost multicast routing [9], and degree-constrained least-cost multicast routing [10], has not always considered the energy consumption. Therefore, these QoS multicast routing schemes cannot be directly used in MANET.

It has been demonstrated that the problem of QoS multicast routing is NP-complete [11]. In the field of artificial intelligence, genetic algorithm is a powerful tool to solve the NP-complete problem. Genetic algorithm is an adaptive heuristic search algorithm based on the evolutionary ideas of natural selection and genetics. The "survival of fittest" filter is applied to the population, which consists of crossover and mutation operators. Crossover exchanges genes of two selected chromosomes to generate an offspring. Mutation randomly modifies some genes of a chromosome to keep the search algorithm away from local optimum and prevent converging too fast. The entire population is ordered on the basis of fitness value of individuals and the best N_p individuals are retained, where N_p is the population size. When generations are produced in an iterative manner, the average fitness of each generation is expected to be improved. The general flow chart of genetic algorithm is shown in Figure 1.

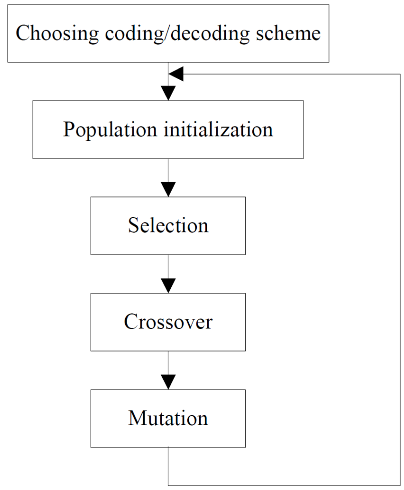

Figure 1. The general flowchart of a genetic algorithm.

In this paper, we study the source-based multicast routing problem. We propose an energy-efficient genetic algorithm to find the delay-constrained multicast tree and reduce the total energy consumption of the tree. Experiments have proven that our algorithm is effective and efficient.

The remainder of this paper is organized as follows. Section 2 introduces the energy consumption in a wireless link. Section 3 states the network model and problem description. Section 4 presents the proposed genetic algorithm. Section 5 analyzes the convergence of the proposed algorithm. The performance of the proposed algorithm is evaluated in Section 6. Section 7 concludes this paper.

2. Background

Compared to wired networks, MANETs have the "wireless multicast advantage" [3], which means that all nodes within the communication range of a transmitting node can receive the multicast message with only one transmission if they are equipped with omni-directional antennas. The models of energy consumption for a link between two nodes are studied in [12]. We generalize the model of the minimum energy needed for a link between nodes v_i and v_j as follows:

$$P_{i,j} = k_1(r_{i,j})^{\beta} + k_2, \tag{1}$$

where $r_{i,j}$ is the Euclidean distance between v_i and v_j, k_1 is a constant dependent on the properties of the antenna, β is the path loss exponent that depends on the propagation losses in the medium, and k_2 is a constant that accounts for the overheads of electronics and digital processing. The energy consumption model $P_{i,j}$ indicates that the communication between two nodes tends to consume less energy if intermediate nodes can be used for forwarding.

3. Network Model and Problem Description

We assume that each node in a MANET determines the distance between itself and its neighbor nodes by using some distance estimation method [13]. The connectivity of the network depends on the transmission power of each node. Each node can dynamically change its transmission power level. A node can use different power level for each multicast tree in which it participates. For simplicity, we assume that all data packets are of the same size. Nodes use omni-directional antennas. Every node v_i in the network has a coverage area CR_i, which depends on the transmission power selected by v_i to transmit its data packets. Let $CR(v_i)$ be the set of nodes within the coverage area of v_i, and r'_{CR_i} denotes the maximum distance between v_i and v_j, where $v_j \in CR(v_i)$. The nodes within CR_i are called as the *neighbors* of v_i.

According to the coverage area of each node, a MANET can be modeled as a graph $G = (V, E)$, where $V = \{v_1, v_2, ..., v_n\}$ is a set of nodes (mobile hosts) and $E = \{(i, j) | v_i, v_j \in V\}$ is a set of links. $(i, j) \in E$ indicates that v_i and v_j are within the coverage area of each other. Each link (i, j) is associated with a delay d_{ij} and a distance l_{ij}. d_{ij} describes the data transmission delay between v_i and v_j, which includes queuing delay and propagation delay. l_{ij} denotes the Euclidean distance between v_i and v_j. Both d_{ij} and l_{ij} are positive real numbers.

Let $s \in V$ be a multicast source and $D \subseteq V - \{s\}$ be a set of destination nodes. A multicast tree $T(s, D) \subseteq G$ is a tree rooted at *s* and reaching all of the destinations in *D*. The delay of a path on *T* from *s* to a node $v_t \in D$, denoted as $delay(p_T(s, v_t))$, is described as:

$$delay(p_T(s, v_t)) = \sum_{(i, j) \in p_T(s, v_t)} d_{ij}. \tag{2}$$

The delay-constrained minimum Steiner tree problem is then to find a minimum cost multicast tree $T^*(s, D)$ that satisfies the following constraint

$$delay(p_T(s, v_t)) \le \delta, \quad \forall v_t \in D, \tag{3}$$

where δ is the overall allowable delay from *s* to a destination $v_t \in D$.

4. The Proposed Multicast Routing Algorithm

4.1. Coding

The representation of candidate solutions is critical for designing a well-performed genetic algorithm. A number of representations for a tree, such as one-dimensional binary code [14], Prüfer numbers [15] and, sequence and topology encoding (ST encoding) [16], have been developed. However, these representations are likely to generate illegal trees (e.g., ST encoding), or have poor locality (e.g., Prüfer numbers), or have low efficiency that the required search space increases remarkably with the increase of the network size (e.g., one-dimensional binary code). Recent studies on network optimization avoid these problems by directly manipulating trees, i.e., using a data structure of a tree to describe the chromosome. With this method, a tree directly represents a chromosome. Therefore, the coding/decoding operations are omitted. In our study, we use the tree structure coding method, in which a chromosome represents a multicast tree directly.

4.2. Initial population

Two issues should be considered in the process of population initialization: (1) population size N_p; (2) the method of population formation. N_p is set by the system. In [17], a random depth-first search algorithm was employed for construction of random multicast Steiner trees. In the proposed algorithm, the formation of initial population is the same as in [17]. The searching process begins at s and randomly selects an unvisited node at each node for next visit. This process terminates when all destinations have been visited.

4.3. Fitness function

Fitness function is used to measure the quality of the individual in the population. The fitness function should reflect the individual performance: the "good individual" has bigger fitness than the "bad one". The definition of the fitness function is as follows:

$$f(T) = \frac{a}{cost(T)} \prod_{v_i \in D} \phi(delay(p_T(s, v_t)) - \delta), \tag{4}$$

where

$$\phi(delay(p_T(s, v_t)) - \delta) = \begin{cases} 1, & \text{if } delay(p_T(s, v_t) - \delta) \le 0, \\ \gamma, & \text{if } delay(p_T(s, v_t) - \delta) > 0. \end{cases} \tag{5}$$

a is the positive real weighting coefficient. δ is the maximum allowable delay from s to v_t, where $v_t \in D$. $\phi(\cdot)$ is a penalty function. The value γ $(0 < \gamma < 1)$ determines the degree of penalty: the smaller the value of γ, the higher the degree of penalty. In our experiments, we set $\gamma = 0.5$.

This paper reduces the energy consumption of a multicast tree to maximize the network service time. Let $B(v_i)$ be a set of immediate succeeding nodes of v_i on *T*. r_i' denotes the maximum distance between v_i and v_j, where $v_j \in B(v_i)$. Thus, the total energy cost of *T* is given as:

$$cost(T) = \sum_{v_i \in T} c_i^T = \sum_{v_i \in T} b[k_1(r_1')^{\beta} + k_2], \tag{6}$$

where c_i^T is the energy cost of v_i, and *b* is the coefficient of the positive real number. Note that the energy cost of leaf nodes is zero. Particularly, we set $k_1 = 1$, $k_2 = 0$, $b = 1$ and $\beta = 2$ in our experiments.

4.4. Selection of parents

In Holland's original genetic algorithm, parents are entirely replaced by new offspring. Offspring may be less fit than their parents since the genetic algorithm are blind. The strategy of fully replacing parent individuals may lose some fitter individuals in the process of evolution. In the proposed genetic algorithm, an elitist model is adopted as the selection operator. First, we select the best individuals and directly copy them to the next generation. Then, we select the rest by the roulette wheel selection model. The probability for selecting a parent T_i, denoted as $p(T_i)$, is given by:

$$p(T_i) = \frac{f(T_i)}{\sum_{j=1}^{N_p} f(T_j)}. \tag{7}$$

Elitist model can improve the performance of genetic algorithm, because it prevents losing the best chromosomes.

4.5. Crossover scheme

Based on the roulette wheel selection, a pair of chromosome is selected as the parents to produce a single offspring. Let T_a and T_b be the selected parents. The crossover operator used in our work is an improved scheme proposed in [9]. The crossover operator generates a child T_c by identifying the same links between T_a and T_b, and retaining these common links in T_c. According to the definition of fitness function, the "better" individual has higher probability of being selected as a parent. Thus, the common links between two parents are more likely to represent the "good" traits. However, retaining these common links in T_c may generate some separate sub-trees. Therefore, links are needed to be selected to connect these sub-trees into a multicast tree.

The process of connecting separate sub-trees is as follows. First, two separate sub-trees are randomly selected among these sub-trees. Then, the selected sub-trees are connected by the least-delay path to form a new sub-tree. The connecting process repeats until a multicast tree is constructed. In order to find the least-delay path between two sub-trees, we add two nodes. One node is connected to all of the nodes of one sub-tree with links which have zero delay associated with them. Similarly, the other node is connected to all the nodes of the other sub-tree with zero-delay links. Hence, the least-delay path between two sub-trees is the least-delay path between two added nodes. Clearly, there are no routing loops in the multicast tree with this connecting method.

An example of crossover procedure is shown in Figure 2. The same links of T_a and T_b are retained in T_c. Then, all sub-trees are connected with least-delay paths which are denoted as dot lines in T_c.

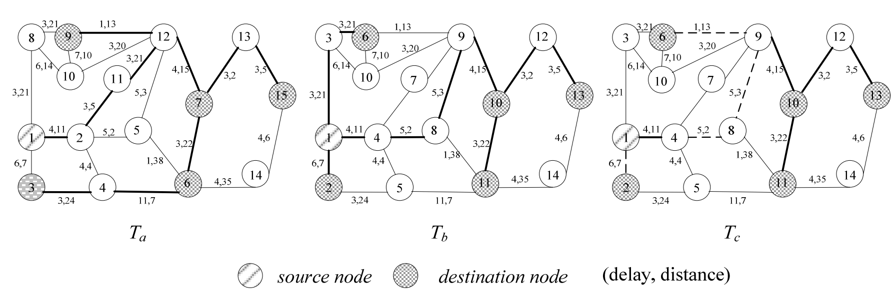

Figure 2. Example of crossover operation.

4.6. Mutation

When a new offspring is produced, the mutation operation is performed according to the mutation probability p_m. First, mutation procedure randomly selects a subset of nodes and breaks the multicast tree into some separate sub-trees by removing all the links that connected these selected nodes and their farthest child node on *T*. Then, it re-connects these separate sub-trees into a new multicast tree with least-delay paths.

Figure 3 shows the flowchart of the proposed genetic algorithm. The corresponding pseudo code is given in Figure 4.

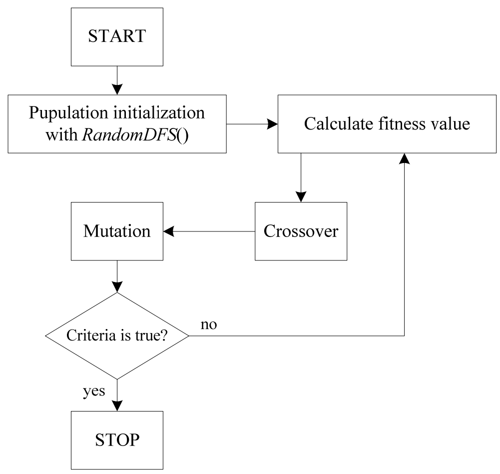

Figure 3. The flowchart of the proposed genetic algorithm.

Proposed_GA (G, s, D)

{

1. *for* $(i = 1; i <= N_p; i{+}{+})$ { // N_p : population size

2. *Chromosome*(i) = *Random DFS*(G, s, D); // *RandomDFS*(): random depth-first search algorithm

}

3. *for* $(j = 1; j <= N_g; j{+}{+})$ { // N_g : the number of generations

4. *select the best individuals and copy them into the next generation*;

5. $for\ (k = 1;\ k <= N_p - N_{optimal};\ k{+}{+})\ \{$ // $N_{optimal}$: the number of the best individuals

6. $T_a = MSTSelect\ (Chromosome)$

7. $T_b = MSTSelect\ (Chromosome)$

8. $T_c = Crossover\ (T_a, T_b);$

9. $if\ (rand() < p_m)$ // p_m : mutation probability

10. $Mutation\ (T_c);$

}

}

11. *Select the best individual and output it*;

}

Figure 4. The pseudo code of the proposed genetic algorithm.

5. Analysis of the Proposed Algorithm

The characteristics of the proposed algorithm are as follows: (1) crossover probability $p_c \in (0, 1]$; (2) mutation probability $p_m \in (0, 1)$; and (3) adopting the elitist model for selection. According to Theorem 2.7 in [18], the proposed algorithm could finally convergence to the global optimal solution.

For a large-scale network, it is time-consuming to obtain the optimal solution to the multicast routing problem with multiple QoS constraints, which is NP-complete. Furthermore, genetic algorithms may not be promising candidates for supporting delay-sensitive applications in MANETs because they may involve a large number of iterations. This problem can be overcome by the hardware implementations of genetic algorithms (e.g., field-programmable gate array (FPGA) chips) [19], which are very fast. In addition, genetic algorithms are not very sensitive to network size [20]. In this regard, the proposed genetic algorithm is quite promising for multicast routing in MANETs.

6. Experiments

We have implemented the proposed genetic algorithm in MS VC++ 6.0 using the genetic algorithmlib which is a C++ Library of Genetic Algorithm. The experiments were performed on a PC with Pentium Dual-core 2.5GHz CPU (2 GB memory). In all experiments, the mutation probability is set to 0.05 (a typical mutation option) and the crossover probability is set to 1. The population size N_p is set to 15.

6.1. Results for fixed network

These experiments mainly test the convergence ability and successful ratio. The successful ratio (*SR*) which is defined as

$$SR = \frac{\text{number of request successfully routed}}{\text{total number of routing requests}}. \tag{8}$$

When the multicast tree constructed by the algorithm satisfies the delay constraint, the routing request is considered as successfully routed one. These experiments mainly involve the deterministic, weighted network topology (15 nodes) depicted in Figure 2. Node 1 is the source and nodes 3, 6, 7, 9, 15 are the destinations.

Figures 5 and 6 show the convergence processes of the proposed algorithm under $\delta = 18$ and $\delta = 25$, respectively. As shown in Figures 5 and 6, the proposed algorithm can converge to the global solution quickly.

Figure 7 shows the comparison of *SR* between the proposed algorithm and least delay multicast tree algorithm (LDT) which has the highest *SR*. The result is based on 1000 randomly generated routing requests. The source and destinations of each request are generated randomly. As shown in Figure 7, it is obvious that the two algorithms have the same *SR*. This proves that the proposed algorithm can find the feasible multicast tree if one exists.

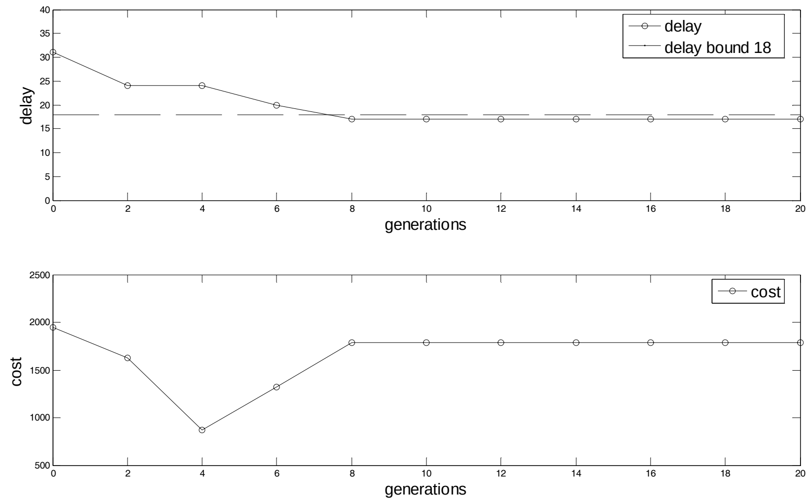

Figure 5. Convergence process of the proposed genetic algorithm with $\delta = 18$.

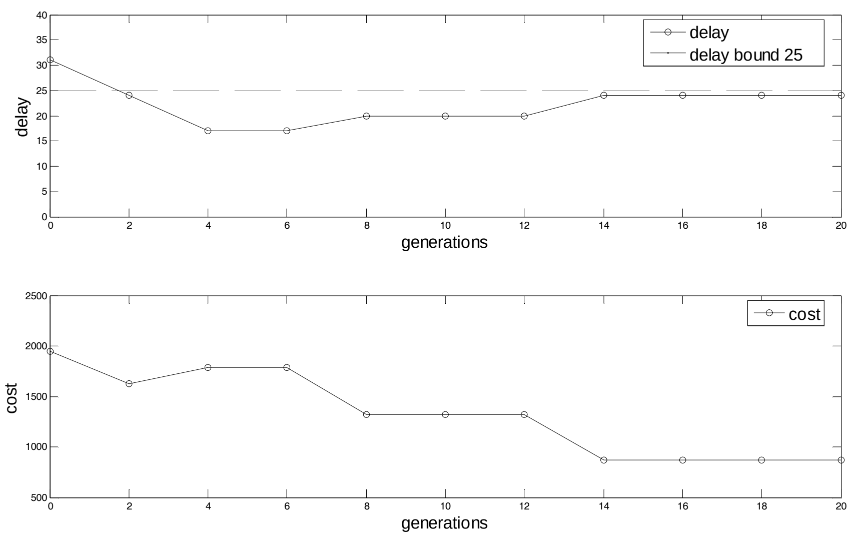

Figure 6. Convergence process of the proposed genetic algorithm with $\delta = 25$.

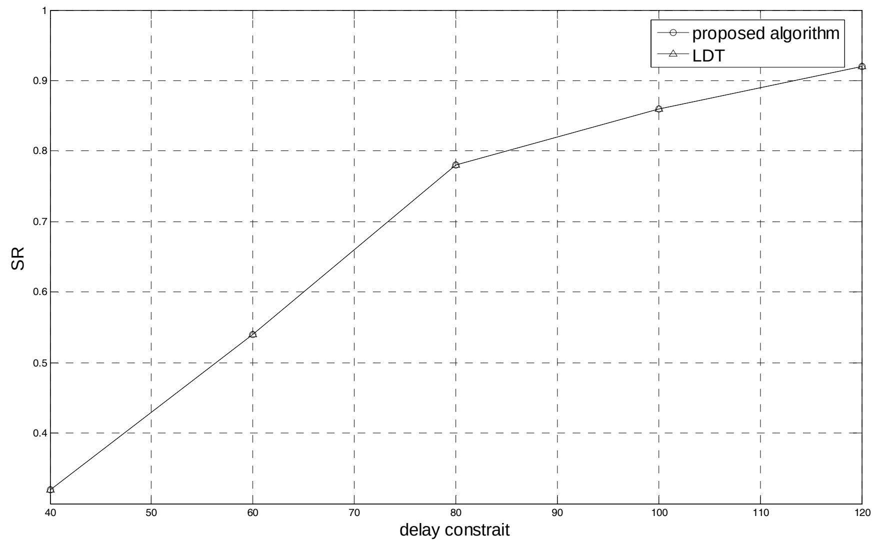

Figure 7. Comparison of *SR* between the proposed genetic algorithm and LDT.

6.2. Results for random networks

In this section, we test the running time. The simulation studies involve random networks with 20-100 nodes. The distance of each link is uniformly distributed in [10, 200] and the delay of each link in [0, 50]. The maximum allowable delay δ is uniformly distributed in [30, 160]. For each request, the source and destinations are randomly generated. MANETs can be used in applications including military battlefield (e.g., moving platoon or company), rescue missions, conference room and so on [21]. These applications involve networks with sizes that range from small to medium, e.g., tens of nodes. Simulations reflect this practical reality. For larger networks, some cluster based schemes can be used.

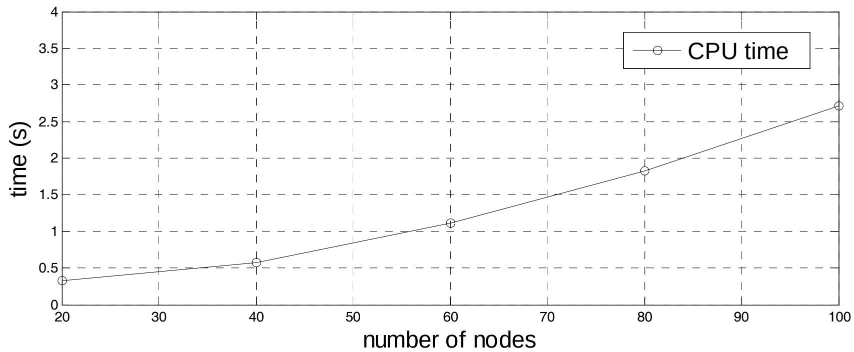

Figure 8. Running time of the proposed genetic algorithm.

Figure 8 shows the experimental results obtained by the proposed algorithm. From Figure 8 we can see that the running time of the proposed algorithm grows slowly with the size of the network. Even for the network with 100 nodes, the running time is fairly desirable.

7. Conclusions

Power awareness is crucial in a mobile wireless network, particularly in a MANET. Nodes need to reduce their power consumption to prolong their battery lifetime. In this paper, we proposed the energy-efficient delay-constrained multicast routing algorithm. The proposed algorithm is a source-based algorithm which takes into account energy consumption as well as end-to-end delay in route selection. The proposed algorithm applies crossover and mutation operations directly on trees, which simplifies the coding operation and omits the coding/decoding process. Heuristic mutation technique can improve the total energy consumption of a multicast tree. A series of experiments was performed to verify the convergence performance, *SR* and running time of the proposed algorithm. The results demonstrate that the proposed algorithm is effective and efficient.

References

[1] S. Sesay, Z. Yang and J. He, A survey on mobile ad hoc wireless network, Inform. Tech. J. 3 (2004), 168-175.

[2] S. Singh, M. Woo and C. S. Raghavendra, Power-aware routing in mobile ad hoc networks, Proc. the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking, October 1998, pp. 181-190.

[3] J. E. Wieselthier, G. D. Nguyen and A. Ephremides, On the construction of energy-efficient broadcast and multicast trees in wireless networks, Proc. IEEE INFOCOM 2000, (2000), 585-594.

[4] J. E. Wieselthier, G. D. Nguyen and A. Ephremides, Distributed algorithms for energy-efficient broadcasting in ad hoc networks, Proc. IEEE Military Communications Conference, 2002, pp. 820-825.

[5] M. Cagalj, J. P. Hubaux and C. Enz, Minimum-energy broadcast in all wireless networks: NP-completeness and distribution issues, Proc. ACM International Conference on Mobile Networking and Computing, 2002, pp. 172-182.

[6] A. Misra and S. Banerjee, MRPC: maximizing network lifetime for reliable routing in wireless environment, Proc. IEEE Wireless Communications and Networking Conference, 2002, pp. 800-806.

[7] B. Wang and Sandeep K. S. Gupta, Extending the lifetime of multicast trees in WANETs, J. Inform. Sci. Eng. 20(3) (2004), 425-447.

[8] A. Tawfig and Z. Taieb, Delay-constrained, low-cost multicast routing in multimedia networks, J. Para. Distri. Comput. 61(9) (2001), 1307-1336.

[9] Z. Wang, B. Shi and E. Zhao, Bandwidth-delay-constrained least-cost multicast routing based on heuristic genetic algorithm, Comp. Comm. 24 (2001), 685-692.

[10] S. Y. Tseng, Y. M. Huang and C. C. Lin, Genetic algorithm for delay-and degree-constrained multimedia broadcasting on overlay networks, Comp. Comm. 29 (2006), 3625-3632.

[11] Z. Wang and J. Crowcroft, Quality of service for supporting multimedia applications, IEEE J. Selected Areas Comm. 14(7) (1996), 1228-1234.

[12] L. M. Feeney and M. Nilsson, Investigating the energy consumption of a wireless network interface in an ad hoc networking environment, Proc. IEEE INFOCOM 2001, (2001), 1548-1557.

[13] W. C. Y. Lee, Mobile Communication Engineering, McGraw-Hall, 1993.

[14] F. Xiang, L. Junzhou, W. Jieyi and G. Guanqun, QoS routing based on genetic algorithm, Comp. Comm. 22 (1999), 1392-1399.

[15] A. T. Haghighat, K. Faez, M. Dehghan, A. Mowlaei and Y. Ghahremani, A genetic algorithm for Steiner tree optimization with multiple constraints using Prüfer number, Proc. Eur Asia-ICT 2002, October 2002, pp. 272-280.

[16] Y. S. Yen, Y. K. Chan, H. C. Chao and J. H. Park, A genetic algorithm for energy-efficient based multicast routing on MANETs, Comp. Comm. 31 (2008), 2632-2641.

[17] C. P. Ravikumar and R. Bajpai, Source-based delay-bounded multicasting in multimedia networks, Comp. Comm. 21(2) (1998), 126-132.

[18] C. Guoliang, W. Xufa, Z. Zhenquan and W. Dongsheng, Genetic Algorithm and its Application, People's Posts and Telecommunications Press, 1996.

[19] S. D. Scott, A. Samal and S. Seth, HGA: a hardware-based genetic algorithm, Proc. FPGA 95, February 1995, pp. 53-59.

[20] G. Tufte and P. C. Haddow, Prototyping a GA pipeline for complete hardware evolution, Proc. EH 99 (1999), 18-25.

[21] C. W. Ahn, R. S. Ramakrishna and C. G. Kang, Efficient clustering-based routing protocol in mobile ad-hoc networks, Proc. IEEE Vehicular Technology Conference, September 2002, pp. 1647-1651.