Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2015, Article ID 258343, 7 pages
http://dx.doi.org/10.1155/2015/258343

Research Article

A Genetic Algorithm Based Minimal Maximal Load Tree Routing Algorithm in Wireless Sensor Networks

Zhi Huang

School of Transportation & Traffic Management, Mianyang Normal University, Mianyang 621000, China

Correspondence should be addressed to Zhi Huang; huangzhi2003@126.com

Received 25 June 2015; Revised 7 October 2015; Accepted 18 October 2015

Academic Editor: Lu Liu

The key issue of static routing algorithms is how to construct an energy efficient routing tree that is utilized during the whole network duration in order to extend network lifetime. In this paper, we have illuminated that, in applications that define network lifetime as the time when the first sensor dies, the optimal routing tree should be the routing tree with minimal maximal load of all sensors and named such trees the Minimal Maximal Load Tree (MMLT). Since the procedure of constructing a routing tree is complex and the number of possible routing trees in a network is very huge, we have proposed a genetic algorithm (GA) based algorithm to obtain approximate Minimal Maximal Load Tree (MMLT). Each individual corresponds to a routing tree, and the fitness function is defined as the maximal load of all sensors in accordance with the routing tree that the individual corresponds to. Thus, approximate MMLT is obtained and network lifetime is extended. Simulation results show that our proposed algorithm notably extends network lifetime.

1. Introduction

Wireless sensor networks consist of homogeneous, tiny sensors that can measure the environment nearby and report sensed data to the sink. With the development of computer, communication, electronic, and other technologies, wireless sensor networks have become an important technology and have been widely applied in many fields, such as military, environment, and space exploration [1–7]. In wireless sensor networks, all sensors are powered by lightweight batteries. Therefore, limited energy of sensors constrains network lifetime and is a serious bottleneck. And how to improve energy efficiency, thereby extending network lifetime, is a critical issue in the study of wireless sensor networks [7–10].

Many energy efficiency routing algorithms have been proposed in order to improve energy efficiency and extend network lifetime. These proposed algorithms fall into two types: static routing algorithms and dynamic routing algorithms. With the former, only one routing is constructed and it is used during the whole network duration. With the latter, multiple routings are constructed and used based on changing status of a network. Although dynamic routing can respond to changing status of a network, it needs to obtain current status of the network, such as all sensors' current residual energy, which requires that networks meet some specific conditions. For example, with Ratio-w [11] or Sum-w [11], it is assumed that there is enough time between two times of data gathering so that all sensors can report their current energy to the sink [11]. Therefore, dynamic routings are only suitable for the networks meeting some specific conditions and not universally suitable for all networks. On the contrary, static routings usually do not require networks to meet specific conditions and can be applied to all applications because they need not obtain changing status of a network.

Many static routing algorithms have been proposed. The key issue of static routing algorithms is how to construct a routing tree that is utilized during the whole network duration. Usually, the energy consumed to deliver a data packet between two sensors is set as the weight of the link between the two sensors [11, 12]. PEDAP (Power Efficient Data Gathering and Aggregation Protocol) [12] constructs a routing tree based on MST (Minimum Spanning Tree) algorithm. However, it is pointed out that the optimal routing tree is not MST, but LET (Least Energy Tree) [12]. LET

constructs a minimum path tree with the sink as the root. Therefore, each sensor sends its data packets to the sink with the least energy consumption and the energy consumption of the whole network is also minimal [11].

Network lifetime of wireless sensor networks has two definitions in two kinds of applications: (1) in some applications, network lifetime is defined as the time when the first sensor depletes its energy and dies because the network loses its efficacy even if only one sensor dies and (2) in other applications, the fact that one or a few sensors die has no effect on the application. And network lifetime is defined as the time when a certain proportion of sensors die because only this leads to the fact that the network loses its efficacy [3].

With static routing algorithms, each sensor consumes the same energy in each data gathering because the routing is constant. We define a sensor's energy consumption in each data gathering as the sensor's load. For an application which defines its network lifetime as the time when the first sensor dies and adopts static routing algorithms, its network lifetime is decided not by energy consumption of the whole network but by the load of the sensor with the maximal load because the first dead sensor must be the one with the maximal load. Therefore, the optimal routing tree is the routing tree minimizing maximal load of all sensors, which is named Minimal Maximal Load Tree (MMLT) by us. However, it is very difficult to find the MMLT in a network. This is because the process of constructing a routing tree is complex and most networks have large numbers of possible routing trees, especially when a network has quite a few sensors.

In this paper, we have proposed an algorithm that finds approximate MMLT. The genetic algorithm (GA) is adopted in our algorithm to find approximate MMLT because the GA is a stochastic searching and optimizing technology and appropriate to solve complex problems having huge numbers of possible solutions [9]. In detail, all sensors are divided into different layers according to their minimal hops to the sink. For a sensor, its neighbor sensors in its next layer closer to the sink are its possible parent sensors. A routing tree is constructed as an individual of the GA after each sensor is designated a possible parent sensor. And the objective function of the GA is minimizing maximal load of all sensors. Thus, an approximate MMLT can be obtained. Although routing trees obtained by our proposed algorithm are only approximate MMLT, they still can improve energy efficiency and prolong network lifetimes compared to traditional rooting algorithms because they decrease the maximal load of all sensors.

The remainder of this paper is organized as follows: in Section 2, we describe related works of routing algorithms; in Section 3, we present the model of our work; in Section 4, our proposed algorithm is described minutely; experimental results are provided and analyzed in Section 5; finally, this paper is concluded and future work is presented in Section 6.

2. Related Works

In this section, we summarize some routing algorithms that have been proposed to extend network lifetime.

As mentioned before, routing algorithms in wireless sensor networks have been widely studied and many energy efficient routing algorithms have been proposed in order to extend network lifetime. In this section, we summarize some of these routing algorithms. We also summarize some applications of the GA in the study of routing in wireless sensor networks.

Some routing algorithms are proposed based on Flooding or Gossiping [13, 14]. For example, a gossip-based routing algorithm was proposed in [14]. With this routing, for a route quest received by a sensor the first time, the sensor broadcasts it with the probability p and discards it with the probability $1-p$; and for a route quest received by the sensor not the first time, the sensor discard it [14]. Routing algorithms based on Flooding or Gossiping produce and forward too much route quests that waste sensors' energy and limit network lifetime [11].

In [12], PEDAP (Power Efficient Data Gathering and Aggregation Protocol) was proposed. PEDAP computes a routing tree with MST (Minimum Spanning Tree) algorithm by setting the energy cost delivering a data packet between two sensors as the weight of the link between them [11, 12]. However, routing trees constructed with MST are not optimal energy efficient routing trees and cannot obtain long network lifetime [11].

Based on PEDAP, another algorithm, PEDAP-PA (Power Efficient Data Gathering and Aggregation Protocol-Power Aware) [12], was proposed. With PEDAP-PA, the sink periodically estimates each sensor's residual energy based on the current routing and computes the weight from sensor i to sensor j as follows:

$$w(i,j) = \frac{c(i,j)}{e_i} \tag{1}$$

(see [12]), where $c(i,j)$ is the energy consumption sending a packet from sensor i to sensor j and e_i is the residual energy of sensor i. And routing trees are periodically constructed with the MST algorithm [12]. However, it is a dynamic routing algorithm and not suitable for all applications.

In [11], it is pointed out that the optimal routings are not the routings constructed with MST algorithm, but the routings constructed with LET (Least Energy Tree) algorithm which constructs a minimum path tree with the sink as the root [11]. And two dynamic routing algorithms, Ratio-w and Sum-w, were proposed. It is assumed that there is enough time between two times of data gathering. Before each round of data gathering, all sensors report their residual energy to the sink, and the weight between sensor i and j is computed as follows:

$$\begin{aligned} w(i,j) &= [c(i,j)]^{\alpha}\left[\frac{1}{e_i}\right]^{\beta}, \\ w(i,j) &= \lambda c(i,j) + \delta\left[\frac{1}{e_i}\right] \end{aligned} \tag{2}$$

(see [11]), where the former is used in Ratio-w and the latter is used in Sum-w. And then a routing is constructed with LET

algorithm [11]. However, these two algorithms are dynamic routing algorithms and cannot be applied to all applications.

In [9], a GA based algorithm for energy efficient routing in two-tiered networks is proposed. The bottom tier is sensors and the upper tier is relay nodes with higher power. Relay nodes act as cluster headers, receive data from sensors, and forward the data to the sink through the network formed among them. Network lifetime is mainly determined by relay nodes. The proposed algorithm schedules the data gathering of relay nodes with GA and notably extends network lifetimes. The algorithm is designed for networks with relay nodes, while our proposed algorithm is designed for networks without relay nodes.

In [15], GA is applied to select cluster headers from sensors. In detail, the whole network is divided into some areas, and some sensors in each area act as cluster headers. Cluster headers in each area form a chromosome. Then the most suitable solutions of cluster heads in each area for each round are obtained with GA. The algorithm increases network lifetime. However, it is only suitable for networks with clustering hierarchy.

In [16], GA is applied to solve the optimization problems of finding combinations of sensors detecting and/or locating targets. The algorithm decreases the energy consumption of the whole network. In [17], GA is applied to allocate different detection methods to different sensors in order to increase detection probability. However, the two algorithms are designed and suitable for the applications of detecting targets.

3. The Model

We have considered a square area in which some homogeneous sensors are randomly and evenly deployed, and one fixed sink is deployed at the center of the area. The sensors are powered by lightweight batteries and have limited energy. And the sink is assumed to have limitless energy, memory, and computing power. After the network is deployed, sensors obtain their positions by GPS or other technologies and report their positions to the sink. Therefore, the sink has the data of each sensor's position [18].

The first order radio model [19] is adopted as our radio model. The energy consumed by a sensor to receive k bits is

$$C_R(k) = kE_{\text{elec}} \tag{3}$$

(see [11, 19]), where E_{elec} is the energy consumed by transmitter or receiver circuits to send or receive one bit. The energy consumed by a sensor to send k bits is

$$C_T(k, d) = kE_{\text{elec}} + kE_{\text{amp}}d^{\gamma} \tag{4}$$

(see [11, 19]), where d is the distance between the sending sensor and the receiving sensor, $E_{\text{amp}}d^{\gamma}$ is the energy consumed by transmit amplifier to send one bit, and γ is called path loss exponent with the value 2, 3, or 4. Therefore, the total energy consumed by sensor i to send k bits and sensor j to receive k bits is

$$C(k, d) = 2kE_{\text{elec}} + kE_{\text{amp}}d^{\gamma} \tag{5}$$

(see [11, 19]).

4. A Genetic Algorithm Based Minimal Load Tree Routing Algorithm

As mentioned before, the key issue of static routing algorithms is how to construct a routing tree that is utilized during the whole network duration. And usually, the energy consumed to deliver data packets between two sensors is set as the weight of the link between the two sensors. Some routing algorithms, such as PEDAP (Power Efficient Data Gathering and Aggregation Protocol), are based on MST (Minimum Spanning Tree), that is, construct routing trees by Prim algorithm [12]. However, these routing trees are not optimal routing trees because they do not aim at reducing energy consumption of the whole network. Therefore, routing algorithms based on LET (Least Energy Tree) are proposed. These algorithms, such as Sum-w and Ratio-w, construct a minimum path tree with the sink as the root by *Dijkstra* algorithm. Therefore, all sensors send their data packets to the sink with the least energy consumption and energy consumption of the whole network is also minimal [11].

In [9], it is pointed out that network lifetime is mainly determined by the maximal energy consumption of all relay nodes in each data gathering in two-tiered networks [9]. Similarly, for applications with their network lifetime as the time when the first sensor dies and adopts static routing algorithms, their network lifetime is decided not by energy consumption of the whole network but by the load of the sensor with the maximal load because the first dead sensor must be the one with the maximal load, where the load is defined as a sensor's energy consumption in each data gathering. For example, a small network is shown in Figure 1, in which node 0 is the sink. In the figure, the number beside a link is the weight of the link (i.e., the energy consumed to deliver a data packet between two sensors) and the number beside a sensor is its load, and to simplification, only energy consumed to send data packets is considered because the energy consumed to receive data packets is little, and it assumed that each sensor generates only one data packet in each data gathering. (1) is the network's topology and (2) is the LET constructed by *Dijkstra* algorithm. As can be seen in the figure, (2) obtains the least energy consumption of the whole network, and its maximal load is $(1+1+1+1)\times 4 = 16$, at sensor 2. However, in (3), the maximal load is $(1+1+1)\times 4 = 12$, at sensor 2. Therefore, (3) has smaller maximal load and can obtain longer network lifetime than (2).

Therefore, for applications with their network lifetime as the time when the first sensor dies and adopts static routing algorithms, optimal routing trees are the ones minimizing maximal load of all sensors and we named such routing trees Minimal Maximal Load Tree (MMLT). However, it is very difficult to find MMLT because the procedure of constructing a routing tree is complex and the number of possible routing trees in a network is very huge, especially when the network contains many sensors. In this paper, we applied the genetic algorithms (GA) to find MMLT because the GA is a stochastic searching and optimizing technology and appropriate to solve complex problems having huge numbers of possible solutions [9]. Although routing trees obtained with our proposed algorithm are only approximate MMLT, they aim

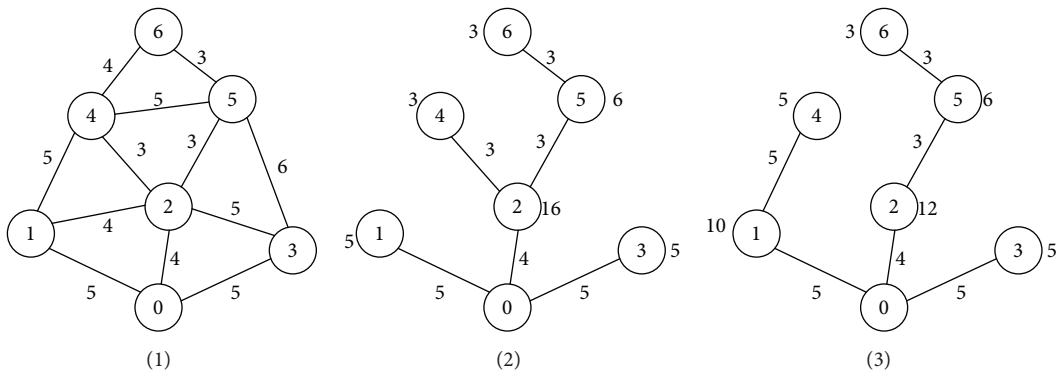

Figure 1: A small network in which node 0 is the sink. (1) The network's topology, (2) the LET, and (3) the routing tree with smaller maximal load than LET [11].

```
Generate an initial population;
Compute the fitness function value of each individual;
n = 0;
While (n < n_gen)
{
   Perform selection;
   Perform crossover and generate new offspring;
   Perform mutation;
   Compute the fitness function value of each individual;
   n++;
}
```

Algorithm 1: The procedure of the GA [9, 20].

at decreasing the maximal load of all sensors, improve energy efficiency, and prolong network lifetimes.

The symbols used in our proposed algorithm are listed as follows:

n_gen: the number of generations in the GA,

n_pop: the size of population in the GA,

pc: the probability that an individual is chosen to perform crossover in the GA,

pm: the probability that an individual is chosen to perform mutation in the GA,

i: an individual in the GA,

s: the sink,

v: a sensor,

t_0: the tree constructed with the *Dijkstra* algorithm when all weights of links are set 1,

$h(v)$: the number of hops from sensor v to s in t_0,

$n(v)$: the set of neighbor sensors of v in t_0,

$p(v)$: the set of possible parent sensors of v in t_0,

LT: the network lifetime,

e: initial energy of sensors,

$L_{\max}$: the maximal load of all sensors.

And the procedure of the GA is shown in Algorithm 1.

Sensor	1	2	3	4	5	6
Parent sensor	0	0	0	1	2	5

Figure 2: The individual corresponding to the routing tree in Figure 1(3) [9].

4.1. Genetic Coding. Each individual corresponds to a routing tree. And each gene corresponds to a sensor's parent sensor. First, the weights of all links are set 1 and the tree t_0 is constructed with the *Dijkstra* algorithm. For each sensor v, its number of hops to the sink in t_0, $h(v)$, can be obtained. And all sensors are divided into different layers in accordance with their values of $h(v)$. Then, the set $p(v)$ can be obtained by the following:

$$p(v) = \{x \mid x \in n(v), h(v) - h(x) = 1\}. \quad (6)$$

For sensor v, it is randomly designated a sensor from $p(v)$ as its parent sensor. Obviously, since each sensor is designated a sensor from its closer layer to the sink as its parent sensor, a routing tree is constructed after all sensors are designated. Thus an individual is generated. For example, the routing tree in Figure 1(3) corresponds to the individual in Figure 2.

4.2. Fitness Function. The fitness function is defined as the maximal load of all sensors. For individual i, a routing tree can be constructed based on it. Then, it is assumed that each sensor generates a data packet and the data packet is forwarded to the sink according to the routing tree. Energy consumption of each sensor receiving or sending the data packet can be computed with (3) and (4) [9]. After all sensors are assumed to generate and send a data packet to the sink, energy consumption of each sensor during the procedure can be computed, the sensor with maximal energy consumption can be found, and the maximal energy consumption is $L_{\max}$ corresponding to i, also the fitness function value of individual i.

4.3. Initial Population and Selection. First, a population involving n_pop individuals is randomly generated.

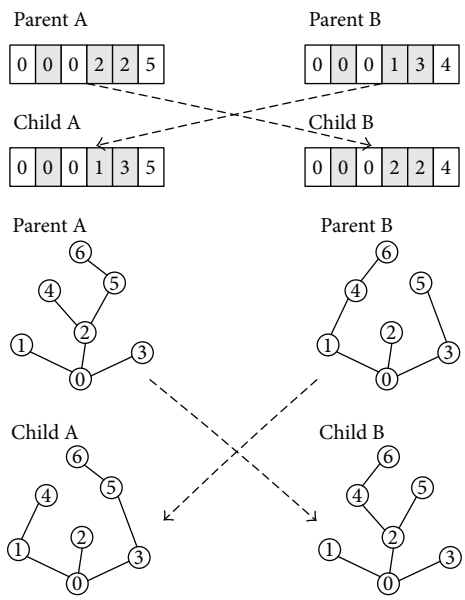

Figure 3: Two individuals performed crossover, where the 2rd, 4th, and 5th genes are chosen to perform crossover [9].

Selection is performed with the Roulette-Wheel or ranking selection method to eliminate inferior individuals and reserve good individuals. In selection, the probability that an individual is selected increases with its value of fitness function.

4.4. Crossover. Each individual is randomly selected to perform crossover with the probability *pc*. Two selected individuals performed crossover to produce new offspring. In this paper, we have adopted the uniform crossover operator [20, 21]. For each gene in the individuals, it is chosen to perform crossover with a probability. For example, two individuals performed crossover in Figure 3, where the 2rd, 4th, and 5th genes are chosen to perform crossover.

4.5. Mutation. Mutation is performed to avoid previous convergence. Each individual is selected with the probability *pm*. For an individual selected to perform mutation, each of its genes is chosen to mutate with a probability [17, 18]. For the sensor corresponding to the gene selected to mutate, v, a new parent is randomly selected from $p(v)$ to perform mutation. An example of mutation is shown in Figure 4, where the 5th gene is selected to perform mutation.

Our proposed algorithm is a centralized algorithm. After the network is deployed, each sensor reports its position to the sink. The sink computes a routing with our GA based algorithm. Then the sink broadcasts the computed routing to the whole network before the network operation. Although the computation of the GA is huge, the computing process does not affect the application because the sink is assumed to have powerful computing capability [18] and the computing process is performed only once before the network operation.

Table 1: The values of the main simulation parameters.

Parameters	Values
Size of the network area	$400 \times 400\ \mathrm{m}^2$
Number of sensors	100/120/140/160/180/200
Communications of sensors	80 m
Initial energy of sensors	10.0 J
Data quantity of a data packet	1×10^3 bits/packet
Generating rate of data packets	1 packet/second
E_{elec} in the first-order radio model	50 nJ/bit
E_{amp} in the first-order radio model	100 pJ/bit/m^2
γ in the first-order radio model	2

Table 2: The values of the main parameters in the GA.

Parameters	Values
The number of generations *n_gen*	100
The size of population *n_pop*	5000
The probability that an individual is chosen to perform crossover *pc*	0.80/0.85/0.90/0.95
The probability that an individual is chosen to perform mutation *pm*	0.10/0.20/0.30/0.40

5. Experimental Validation

Simulations are performed with *OMNet++* to validate our proposed algorithm. We have compared the following three static routing algorithms in the experiments:

(1) Power Efficient Data Gathering and Aggregation Protocol (PEDAP) [12], with which routing trees are constructed by MST (Minimum Spanning Tree);

(2) Least Energy Tree (LET) [11], with which routing trees with the sink as the root are constructed by *Dijkstra* algorithm;

(3) our proposed GA based Minimal Maximal Load Tree routing algorithm (GA-MMLT), with which approximate MMLT is constructed with the GA.

The values of the main simulation parameters are shown in Table 1.

The values of the main parameters in the GA are listed in Table 2. We have set the size of population a large number because the number of the possible individuals (i.e., routing trees) is very huge. Additionally, an elitist approach is adopted, with which 5% of the individuals with the best fitness function values are permitted to proceed the next generation [9, 21].

Network lifetime is defined as the time when the first sensor exhausts its energy and dies [3, 11]. We have compared network lifetimes obtained in different networks with different number of sensors. The experimental results are shown in Figure 5, where each dot is the average value of 5 networks. As can be seen in the figure, our proposed GA-MMLT notably increases network lifetime. With GA-MMLT, network lifetime is prolonged by about 66% on average compared to LET and about 273% on average compared to PEDAP. PEDAP or

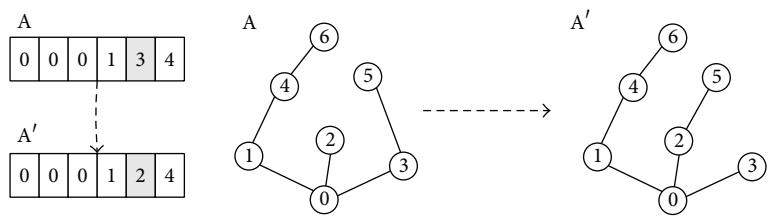

Figure 4: An example of mutation, where the 5th gene is selected to perform mutation [9].

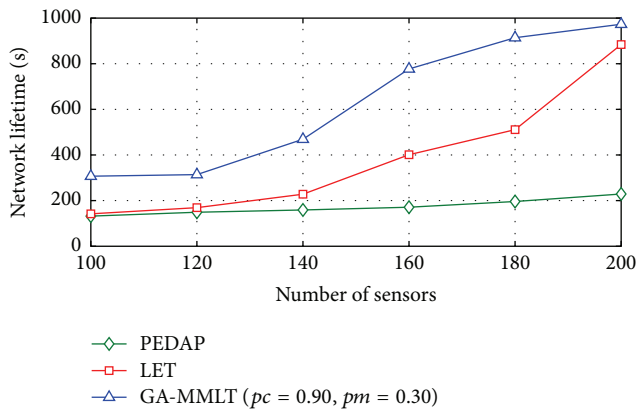

Figure 5: Network lifetime obtained by the three static routing algorithms.

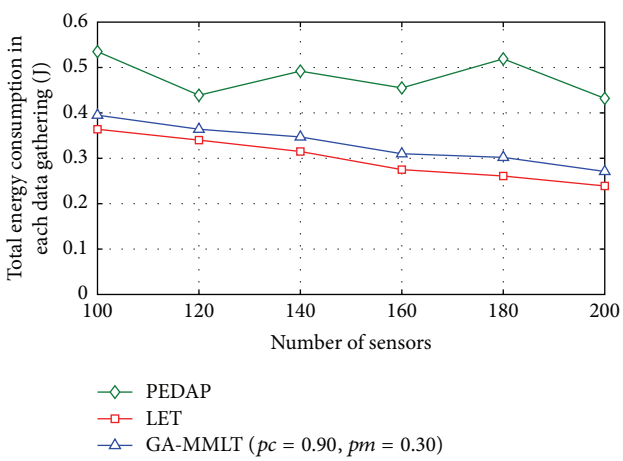

Figure 6: Total energy consumption of the whole network in each data gathering using three static routing algorithms.

LET does not consider the maximal load of all sensors, while GA-MMLT does. Therefore, network lifetimes obtained by PEDAP or LET are shorter than network lifetimes obtained by GA-MMLT which reduces the maximal load of all sensors with the GA.

We have also compared total energy consumption of the whole network in each data gathering using the three static routing algorithms. The experimental results are shown in Figure 6. As can be seen in the figure, LET performs best in this metric, GA-MMLT takes second place, and PEDAP performs worst. This is because LET aims at reducing energy

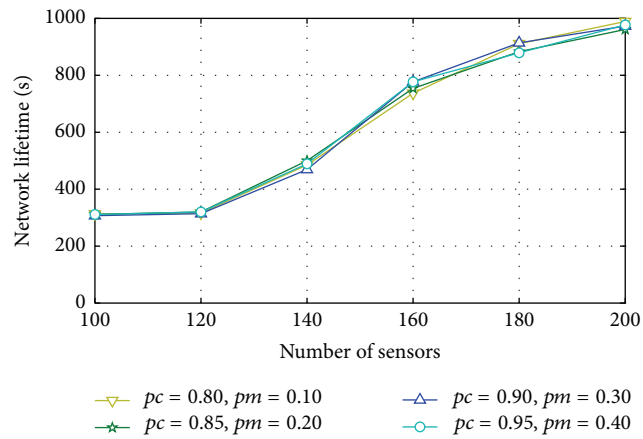

Figure 7: Network lifetimes obtained by GA-MMLT using different values of *pc* and *pm*.

consumption of the whole network. Although GA-MMLT consumes more total energy of the whole network in each data gathering than LET, it obtains longer network lifetimes because it obtains approximate minimal maximal load of all sensors.

Additionally, we have compared network lifetimes obtained by GA-MMLT using different values of *pc* and *pm*. The experimental results are shown in Figure 7. As can be seen in the figure, network lifetimes obtained by GA-MMLT using different values of *pc* and *pm* are very close, where $pc = 0.90$ and $pm = 0.30$ have slight advantage. This shows that GA-MMLT can obtain close performance using different values of *pc* and *pm*.

6. Conclusion and Future Work

In this paper, we have illuminated that, in applications which define network lifetime as the time when the first sensor dies, the key of extending network lifetime is not the energy consumption of the whole network but the maximal load of all sensors and named the routing tree minimizing maximal load of all sensors as the Minimal Maximal Load Tree (MMLT). And furthermore, we have proposed a routing algorithm based on the genetic algorithms (GA) to find approximate MMLT because the procedure of constructing a routing tree is complex and the number of possible routing trees in a network is very huge, especially when the network contains many sensors. In our proposed algorithm, each individual corresponds to a routing tree, and the fitness

function is defined as the maximal load of all sensors in accordance with the routing tree. After some generations of selection, crossover, and mutation, an approximate optimal individual (routing tree) is obtained. Thus, the time the first sensor dies is postponed and network lifetime is extended. For the future work, we plan to propose a novel algorithm that can find the authentic MMLT.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication if this paper.

Acknowledgment

The project was supported by the following scientific research projects of Mianyang Normal University: 2014A07 (no. 07165212) and QD2014A007 (no. 07165211).

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] W. Li, P. Yi, Y. Wu, L. Pan, and J. Li, "A new intrusion detection system based on KNN classification algorithm in wireless sensor network," *Journal of Electrical and Computer Engineering*, vol. 2014, Article ID 240217, 8 pages, 2014.

[3] C.-S. Ok, S. Lee, P. Mitra, and S. Kumara, "Distributed energy balanced routing for wireless sensor networks," *Computers & Industrial Engineering*, vol. 57, no. 1, pp. 125–135, 2009.

[4] K. Akkaya, M. Younis, and M. Bangad, "Sink repositioning for enhanced performance in wireless sensor networks," *Computer Networks*, vol. 49, no. 4, pp. 512–534, 2005.

[5] L. Cui, H. Ju, Y. Miao, T. Li, W. Liu, and Z. Zhao, "Overview of wireless sensor networks," *Computer Research and Development*, vol. 42, no. 1, pp. 163–174, 2005.

[6] C. Vargas-Rosales, J. Mass-Sanchez, E. Ruiz-Ibarra, D. Torres-Roman, and A. Espinoza-Ruiz, "Performance evaluation of localization algorithms for WSNs," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 493930, 14 pages, 2015.

[7] X. Cai, Y. Duan, Y. He, J. Yang, and C. Li, "Bee-Sensor-C: an energy-efficient and scalable multipath routing protocol for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 976127, 14 pages, 2015.

[8] Y. Bi, L. Sun, and N. Li, "BoSS: a moving strategy for mobile sinks in wireless sensor networks," *International Journal of Sensor Networks*, vol. 5, no. 3, pp. 173–184, 2009.

[9] A. Bari, S. Wazed, A. Jaekel, and S. Bandyopadhyay, "A genetic algorithm based approach for energy efficient routing in two-tiered sensor networks," *Ad Hoc Networks*, vol. 7, no. 4, pp. 665–676, 2009.

[10] Y. Fu, Q. Guo, and C. Chen, "A-LNT: a wireless sensor network platform for low-power real-time voice communications," *Journal of Electrical and Computer Engineering*, vol. 2014, Article ID 394376, 19 pages, 2014.

[11] Y. Zhu, D. Shen, W. Wu, Z. Shen, Y. Tang, and Y. Zhu, "Dynamic routing algorithms optimizing lifetime of wireless sensor networks," *Chinese Journal of Electronics*, vol. 37, no. 5, pp. 1041–1045, 2009.

[12] H. Ö. Tan and I. Körpeoglu, "Power efficient data gathering and aggregation in wireless sensor networks," *ACM SIGMOD Record*, vol. 32, no. 4, pp. 66–71, 2003.

[13] S. M. Hedetniemi, S. T. Hedetniemi, and A. L. Liestman, "A survey of gossiping and broadcasting in communication networks," *Networks*, vol. 18, no. 4, pp. 319–349, 1988.

[14] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based ad hoc routing," in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, vol. 3, pp. 1707–1716, IEEE, New York, NY, USA, June 2002.

[15] A. G. Delevar, A. A. Baradaran, and J. Artin, "RGWSN: presenting a genetic-based routing algorithm to reduce energy consumption in wireless sensor network," *International Journal of Computer Science Issues*, vol. 8, no. 5, 2011.

[16] A. L. Buczak, H. Wang, H. Darabi, and M. A. Jafari, "Genetic algorithm convergence study for sensor network optimization," *Information Sciences*, vol. 133, no. 3-4, pp. 267–282, 2001.

[17] Q. Qiu, Q. Wu, D. Burns, and D. Holzhauer, "Lifetime aware resource management for sensor network using distributed genetic algorithm," in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED '06)*, pp. 191–196, Tegernsee, Germany, October 2006.

[18] I. Chatzigiannakis, A. Kinalis, and S. Nikoletseas, "Efficient data propagation strategies in wireless sensor networks using a single mobile sink," *Computer Communications*, vol. 31, no. 5, pp. 896–914, 2008.

[19] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[20] J. Xiong, D. T. Gao, Q. H. Shen, and S. D. Du, "Comparation of crossover operators in genetic algorithm," *Journal of Nanjing University (Natural Sciences)*, vol. 40, no. 4, pp. 432–437, 2004 (Chinese).

[21] L. Davis, *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, NY, USA, 1991.