Traffic-aware routing protocol for wireless sensor networks

Chongmyung Park, Harksoo Kim & Inbum Jung

Cluster Computing
The Journal of Networks, Software Tools and Applications

ISSN 1386-7857
Volume 15
Number 1

Cluster Comput (2012) 15:27-36
DOI 10.1007/s10586-010-0146-3

Cluster Comput (2012) 15:27–36
DOI 10.1007/s10586-010-0146-3

Traffic-aware routing protocol for wireless sensor networks

Chongmyung Park · Harksoo Kim · Inbum Jung

Received: 29 June 2010 / Accepted: 11 November 2010 / Published online: 4 December 2010

Abstract In wireless sensor networks, when a sensor node detects events in the surrounding environment, the sensing period for learning detailed information is likely to be short. However, the short sensing cycle increases the data traffic of the sensor nodes in a routing path. Since the high traffic load causes a data queue overflow in the sensor nodes, important information about urgent events could be lost. In addition, since the battery energy of the sensor nodes is quickly exhausted, the entire lifetime of wireless sensor networks would be shortened. In this paper, to address these problem issues, a new routing protocol is proposed based on a lightweight genetic algorithm. In the proposed method, the sensor nodes are aware of the data traffic rate to monitor the network congestion. In addition, the fitness function is designed from both the average and the standard deviation of the traffic rates of sensor nodes. Based on dominant gene sets in a genetic algorithm, the proposed method selects suitable data forwarding sensor nodes to avoid heavy traffic congestion. In experiments, the proposed method demonstrates efficient data transmission due to much less queue overflow and supports fair data transmission for all sensor nodes. From the results, it is evident that the proposed method not only enhances the reliability of data transmission but also distributes the energy consumption across wireless sensor networks.

Keywords Wireless sensor networks · Routing protocol · Traffic congestion · Genetic algorithm · Energy consumption

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the " "Smart IT Lighting Education Center" support program supervised by the NIPA (National IT Industry Promotion Agency)" (NIPA-2010-C6150-1001-0016).

C. Park · H. Kim · I. Jung (✉)
Department of Computer Information and Communication Engineering, Kangwon National University, Chuncheon, Gangwon, 200-701, Republic of Korea
e-mail: ibjung@kangwon.ac.kr

C. Park
e-mail: cmpark@snslab.kangwon.ac.kr

H. Kim
e-mail: nlpdrkim@kangwon.ac.kr

1 Introduction

Wireless sensor networks are event-driven network systems and are deployed in various fields such as monitoring forest fires, ecology states, machine tools, civil structural strain gauges, and so on. To extend the lifetime of wireless sensor networks, it is necessary for each sensor node to reduce the number of transmissions in the network [1–4]. To satisfy this requirement, sensor nodes usually use a long data sensing cycle when no events occur. However, when an urgent event occurs, a short data sensing cycle is applied to learn detailed information. For example, in the structural strain gauge monitoring system, the sensing rate of 1 packet/sec is enough under the normal operation environment. However, when an abnormal symptom is detected, the sampling rate should be increased for more accurate analyses of the civil structure.

As an additional example, Fig. 1 shows an example of the delivery of events detected by terminal sensor nodes A, B, C, and D. As shown in this figure, when these sensor nodes detect a fire event, the sensing data are delivered to a sink node via the routing path shown by the dotted lines. To collect more information about the fire event, these sensor nodes begin to increase their data sampling rates. The sensing data concentrate on sensor node A to the route to the sink

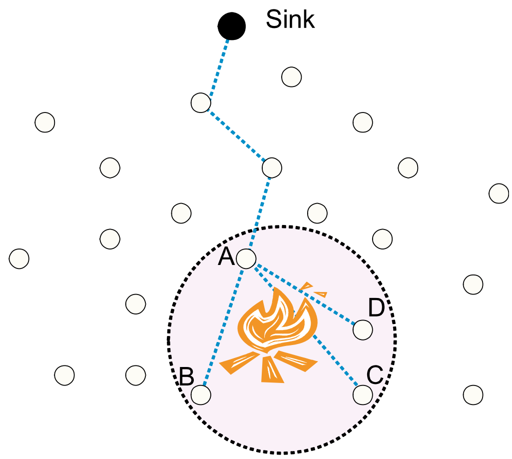


Fig. 1 Event delivery in wireless sensor networks

node. As a result, sensor node A suffers from heavy traffic. This traffic load causes a queue overflow in sensor node A. The queue overflow results in the loss of important information about the urgent event. To avoid traffic congestion, the sensing data generated from nodes B, C, and D should be distributed to other available sensor nodes except sensor node A.

In wireless sensor networks, each sensor node transmits the sensing data by itself and plays the role of router to relay the data of other sensor nodes to a sink node. To accomplish these requirements, sensor nodes usually adopt the queue mechanism to treat the input data packets. However, due to heavy traffic, if the queue handling speed of the sensor node cannot meet the increased input traffic, a queue overflow is inevitable. Since a queue overflow causes data transmission failures, the input data are lost. This phenomenon causes the sink node to fail to learn important data about urgent events. In addition, it hurts the reliability of data transmission. Since the sensor node suffering from heavy traffic has many radio-frequency transmissions, it exhausts the battery energy quickly. This sensor node makes an energy hole in the routing path. Since the wireless sensor networks operate on multiple hops based on relaying the data packet, the energy hole reduces the lifetime of the wireless sensor networks [5].

In this paper, a new routing protocol is proposed to distribute the data traffic congested in a specific sensor node to available neighbor sensor nodes. The proposed method uses traffic awareness and a genetic algorithm. The origin genetic algorithm is often used to find the solution to a complex problem. Due to the limited resources in wireless sensor nodes, the proposed method devises a lightweight genetic algorithm by restricting the number of iteration loops for computation.

The proposed method is aware of the traffic congestion problem in specific sensor nodes and collects information about the child sensor nodes. Based on this information, the lightweight genetic algorithm is performed to select sensor nodes available for forwarding the congested data. Since the heavy traffic loads congested in specific sensor nodes are distributed to the forwarding sensor nodes, the problems due to traffic congestion can be alleviated. In experiments, the proposed method shows efficient data transmission due to much less queue overflow and supports fair data transmission for all sensor nodes. From the experimental data, it is evident that the proposed method not only enhances the reliability of data transmission but also balances the energy consumption distribution across wireless sensor networks. This result contributes directly to the extension of the total lifetime of wireless sensor networks.

The rest of this paper is organized as follows. Section 2 describes wireless sensor network topologies and previous traffic congestion control methods and the genetic algorithm. In Sect. 3, a new routing protocol is proposed to support the efficiency and fairness of data transmission. In Sect. 4, our experimental environments are described, and performance metrics are discussed. In Sect. 5, the performance evaluation is shown. Section 6 concludes the paper.

2 Related work

2.1 Traffic congestion in wireless sensor networks

Since wireless sensor networks work on limited network bandwidth, traffic congestion frequently occurs during the relay process of sensing data. To solve this problem, several studies have been proposed based on the reduction of data sensing rates in terminal sensor nodes [3, 6].

The CODA (COngestion Detection and Avoidance in Sensor Network) proposed 2 cases to detect the traffic congestion phenomenon [6]. In the first case, the sensor nodes in the middle of the routing path monitor their input data traffic. After the input traffic is compared with a specified threshold, the result is relayed to a sink node via multiple hops. In the second case, a sink node monitors its own input data traffic to find traffic congestion. In both cases, when detecting traffic congestion, the sink node uses ACK messages to lower the data sensing rates of the terminal sensor nodes detecting the events.

The ESRT (Event-to-Sink Reliable Transport Protocol) improved the reliability of event data transmission in wireless sensor networks [3]. In this method, all sensor nodes monitor their internal buffers to find traffic congestion. When traffic congestion is detected, the result is sent to the sink node. The sink node broadcasts the congestion state to all sensor nodes. After receiving the broadcast message, each sensor node tries to reduce the data transmission rate. Due to the reduced data traffic, the traffic congestion problem could be alleviated gradually. However, since this approach did not distinguish the event types while undergoing

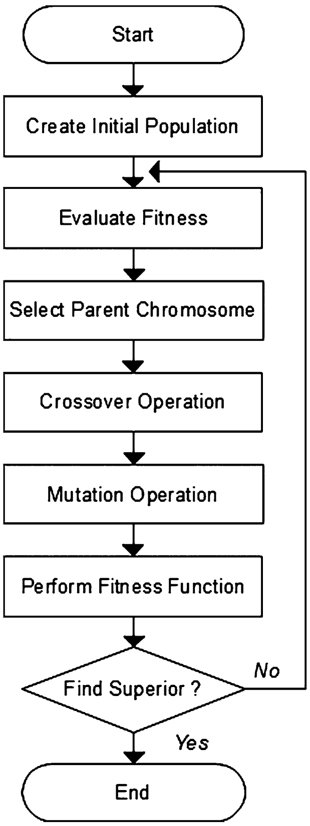


Fig. 2 Genetic algorithm

the traffic congestion, the entire flow of network traffic was restricted, and reliable service for each event type could not be supported.

When traffic congestion occurs, the previous methods required the reduction of data sensing rates in the terminal sensor nodes. In these methods, the sink node did not get enough information to the urgent events. In this paper, instead of the reduction of data sensing rates, the distribution of traffic loads is proposed. Since this approach could provide enough sensing data, the wireless sensor networks make accurate decisions and can respond quickly to the urgent event.

2.2 Genetic algorithm

The genetic algorithm is used to find the solution to complex problems. This algorithm has been applied to machinery learning, robot engineering, TSP problems, various optimal solution problems, and so on [7]. Figure 2 shows the flowchart of the genetic algorithm. First, an initial population of chromosomes is created to represent the solution to complex problems. Second, this algorithm evaluates the fitness of the original chromosomes and selects a parent chromosome to apply genetic operations. Third, the algorithm evolves the population through genetic operations such as crossover and mutation. After that, as this algorithm applies the fitness function to newly created individual objects, only the superior objects remain in the next generation. If a superior object does not exist, this algorithm repeats the loop iteration.

To implement the genetic algorithm, chromosomes for the given problems are designed, and genetic operations are designated. In addition, a fitness function should be devised for target applications. The chromosomes could be represented as the binary denotation or permutation denotation or real integer expression. In this paper, a chromosome is composed of sensor nodes' ID, data transfer rates, and data forwarding rates. To find an optimal solution, the chosen genetic operations are applied to the constituted chromosomes.

3 Traffic-aware routing protocol

When events occur, the data sensing rates of terminal sensor nodes increase, and the relay sensor nodes located in a routing path have increased numbers of data packets. Due to the increased number of data packets, the sensor nodes in the bottleneck suffer from traffic congestion. The traffic congestion causes queue overflow and packet collisions in wireless sensor nodes. Due to these problems, data packets involving information about urgent events may be lost. In addition, the lost data packets in the middle of relay bring out the inefficient use of network resources as well as an unreliable network state. To overcome these problems, the TARP (Traffic-Aware Routing Protocol) is proposed. The proposed method is aware of the congested traffic in each sensor node and uses the genetic algorithm to select data forwarding sensor nodes. Due to the limited resources in wireless sensor nodes, the TARP uses a lightweight genetic algorithm that restricts the number of iteration loops to find a superior object.

3.1 Representation of chromosome

To apply the genetic algorithm, representation of chromosomes is needed first. In the TARP, a structure composed of real numbers is used to represent the chromosomes. The next diagram shows the structure of the chromosomes. In the initialization step, based on the information about neighbor sensor nodes, the chromosomes are created, and solution sets to apply the genetic operations are constituted.

FIT	CID[n]	CR[n]	FR[n]	NID[2][n]	NR[2][n]

- FIT: FITness
- CID: Child node ID
- CR: Child node's data transfer Rate
- FR: Forwarding Rate
- NID: Neighbor node ID
- NR: Neighbor node's data transfer Rate

In this diagram, the FIT(FITness) is calculated via the fitness function. According to the FIT value, an optimal chromosome is selected from the population. The information about selected chromosomes is spread out over the child sensor nodes. Based on this information, the child sensor nodes distribute the traffic loads to their neighbor sensor nodes. The CID and the CR represent the child sensor nodes and their data transfer rate. A chromosome includes information about all child sensor nodes. The NID and NR are the information about neighbor sensor nodes within 2 hops from the child sensor node. Since a chromosome includes information about individual neighbor sensor nodes per each child sensor node, it is possible to build up many solution sets. The FR means the data-forwarding rate. It is initialized from the gap between the current data transfer rate and the predefined threshold value. Based on the FR value and the NID and the NR values, many solution sets can be formed. From these solution sets, an optimal chromosome is chosen to forward the congested traffic.

Figure 3(a) shows the routing tree of wireless sensor networks as an example. Sensor nodes 12, 13, and 14 are the children of sensor node 6. Figure 3(b) represents the chromosome structure of sensor node 6. As shown in the Fig. 3(b), sensor node 6's chromosomes include the data forwarding rates allocated to the children, the data transfer rates of children sensor nodes, and the data transfer rates of the neighboring sensor nodes of children sensor nodes.

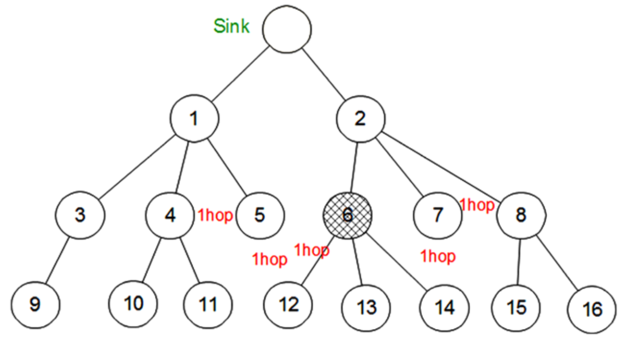
To select the data forwarding sensor node, the TARP uses the data traffic rates of neighboring sensor nodes within 2 hops of a child sensor node. In Fig. 3(a), sensor nodes 5 and 4 are the neighboring sensor nodes within 2 hops of child sensor node 12. Sensor node 5 is within 1 hop of child sensor node 12, and sensor node 4 can be connected within 2 hops from sensor node 12 via sensor node 5. In the Fig. 3(b), we can see sensor nodes 5 and 4 as the NID[1], NID[2] items of this chromosome structure.

This chromosome structure has forwarding rates to each child sensor node as the FR items. The genetic algorithm allocates the FR values in the formative step of the solution sets, and reallocates them through the mutation operations of the genetic algorithm.

The FIT represents the degree of fitness to current data forwarding rates. This field is allocated from executing the fitness function. Based on the FIT value of the chromosomes, the genetic algorithm can distinguish superior chromosomes from the solution sets.

3.2 Fitness function

The TARP manages traffic congestion by changing the network topology. Since it distributes the congested traffic of specific sensor nodes into other sensor nodes, the possibility of data loss due to queue overflow can be greatly reduced.



(a) Routing tree in sensor network.

FIT	CID	CR	FR	NID[1]	NR[1]	NID[2]	NR[2]
-	12	5	2	5	6	4	10
	13	10	4	5	8	4	6
	14	7	3	7	4	8	7

(b) Chromosome structure of sensor node 6.

Fig. 3 Routing tree in sensor network

Thus, the fitness function used in the TARP is designed as in the following equations, (1) and (2). The n is the number of sensor nodes used. The ADT of (1) is the Average Data Traffic of neighbor sensor nodes. It represents the average of the sum of the data transfer rates of neighbor sensor nodes included in a chromosome and their data forwarding rates. The NR is the Neighbor node's data transfer Rate, and the FR is the Data Forwarding Rate.

Equation (2) is the fitness function. It is expressed as an inverse number of the sum of both the standard deviation of the ADT and the ADT itself. Therefore, the fitness is inversely proportional to the average and the standard deviation of the data transfer rates of each sensor node. The large standard deviation means that sensor nodes have uneven data transfer rates. Otherwise, the sensor nodes with a high ADT value signify that their traffic loads are heavy.

Our fitness function uses the data transfer rate as well as the standard deviation. If only the standard deviation is used, the sensor node group with a small standard deviation can be selected as the data forwarding nodes, even if these sensor nodes already have high data transfer rates.

$$\mathrm{ADT} = \sum_{i=1}^{n-1} (\mathrm{NR}_i + \mathrm{FR})/n \qquad (1)$$

$$\mathrm{Fitness}^{-1} = \sqrt{\sum_{i=0}^{n-1} \{\mathrm{ADT} - (\mathrm{NR}_i + \mathrm{FR})\}^2/n} + \mathrm{ADT} \qquad (2)$$

3.3 Genetic operations

The genetic algorithm includes genetic operations such as crossover, mutation, substitution, and so on. As these genetic operations are repeatedly applied to the chromosomes

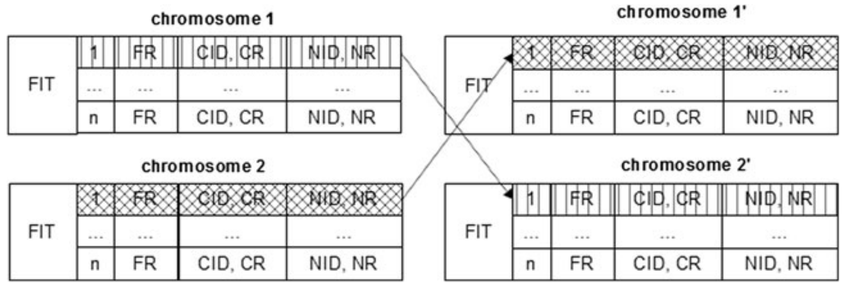


Fig. 4 Crossover operation

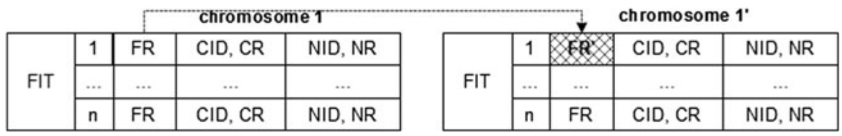


Fig. 5 Mutation operation

included in the solution set, the fitness value of the chromosomes may be changed every generation. In the genetic algorithm, after the fitness values of the chromosomes are compared, only the chromosomes with a high fitness value can remain as superior objects in the solution set. As the genetic algorithm is in progress, the solution set is composed of chromosomes with high fitness values.

Figure 4 represents the crossover operation. This operation creates a new chromosome by exchanging a part of the parent chromosomes. As shown in this figure, two chromosomes (1 and 2) are chosen from a solution set. The CID and CR represent information about the child sensor node. The NID and NR are the information about the neighbor sensor nodes of a child node. These items in a chromosome are exchanged during the crossover process. In this figure, the crossover operation exchanges the first-row items of chromosomes 1′ and 2′. As a result, two new chromosomes, 1′ and 2′, are formed.

Figure 5 represents the mutation operation. If an optimal solution cannot be found in the initial solution set, the mutation operation must be performed for the evolution to succeed. For example, in Fig. 5, the data forwarding rate (FR) of sensor node 1 included in the chromosome 1 is transformed into the new data forwarding rate FR′. Chromosome 1 is mutated into chromosome 1′. According to the mutated FR′ value, sensor node 1 forwards its data traffic to the neighbor sensor nodes while the traffic congestion is in progress. Based on the new FR′ value, the data transfer rates of the neighbor sensor nodes are changed.

3.4 Operation of TARP

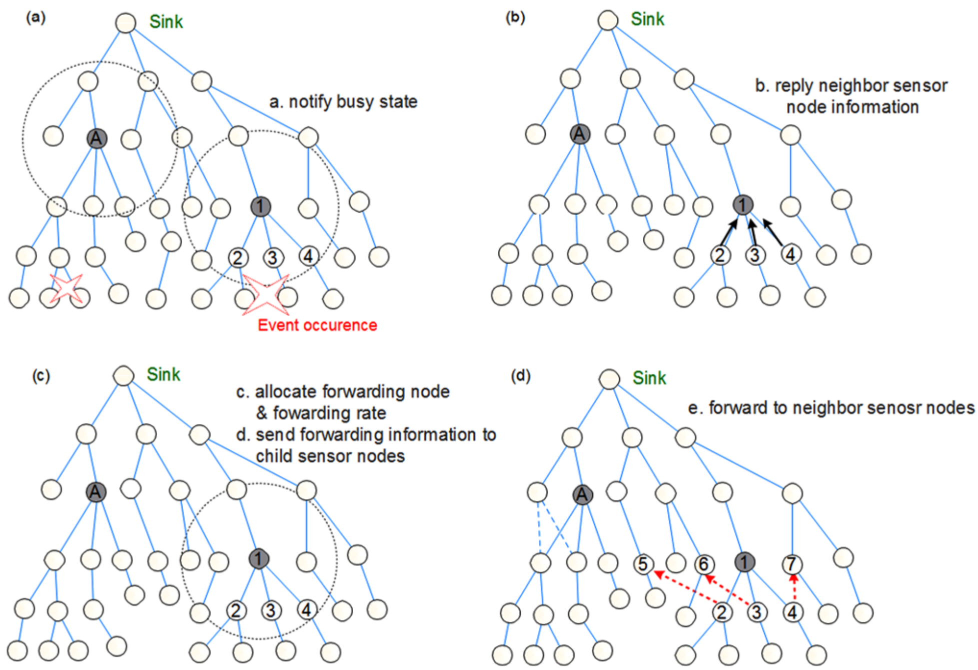
To operate the TARP, each sensor node should be aware of the data transfer rates of the surrounding sensor nodes. Thus, each sensor node broadcasts its own information periodically, and the neighbor sensor nodes store the received information. However, since the re-broadcasting of neighbor sensor node information causes a broadcasting storm, the broadcasting should be limited to the child sensor nodes. In the TARP, each sensor node has information about the sensor nodes located within only 2 hops.

Figure 6 shows an example of the steps of the TARP operation in traffic congestion. Where events take place, the surrounding sensor nodes begin to transfer much of the sensing data to the sink node. For example, we can see that sensor node 1 is located in the bottleneck of the routing path. After sensor node 1 is aware of the traffic congestion, the traffic congestion message is sent to sensor nodes 2, 3, and 4 in order to distribute the input data traffic.

In Fig. 6(b), after sensor nodes 2, 3, and 4 receive the congestion message, they send information about their neighbor sensor nodes to sensor node 1. These neighbor sensor nodes should be located within 2 hops from each child node. In Fig. 6(c), based on the received information, sensor node 1 creates the chromosomes and allocates the data forwarding rate FR. Finally, after the chromosome with the highest fitness is selected, the result is sent to child sensor nodes 2, 3, and 4. Figure 6(d) shows that sensor nodes 5, 6, and 7 are selected as the data forwarding nodes. According to the allocated FR value, since sensor nodes 2, 3, and 4 bypass their data traffic to sensor nodes 5, 6, and 7, the traffic congestion in sensor node 1 is alleviated.

Figure 7 shows the pseudo codes for allocating the forwarding sensor nodes and the forwarding rates. In the initial stage, the genetic algorithm creates the solution sets based on the information of neighbor sensor nodes of child sensor nodes. First, two chromosome objects are selected from the solution sets. As the next step, the crossover operation creates new chromosomes by exchanging the information of two neighbor sensor nodes involved in the chromosomes. The mutation operation transforms the characteristics of the chromosomes by reallocating the forwarding rates. After these genetic operations are applied, the function

Fig. 6 TARP operation




```
allocFWRate_LWGA( ) {          // Lightweight Genetic Algorithm
  initialize()
FOR  i=0 TO end-condition DO   // end-condition is 10 times iteration
select 2 chromosomes
    crossover() // generate new chromosomes with 2 chromosomes
    mutation()     // allocate new data forwarding rate
    evaluate()     // evaluate new chromosomes using the fitness function
  ENDFOR
send()  // notify child nodes of the chromosome with the best fitness
}
```

Fig. 7 Pseudo codes for lightweight genetic algorithm

evaluation calculates the fitness of the newly created chromosomes based on the fitness function. The crossover mutation evaluation operations are performed repeatedly until the end-condition is satisfied. Due to the limited resources in wireless sensor nodes, the TARP uses the lightweight genetic algorithm by restricting the number of loop iterations. As shown in this figure, the iteration is limited to 10 times. Due to the limited iteration loops, the lightweight genetic algorithm generates much less computation burden on the sensor nodes than the original genetic algorithm.

After the loop area is finished, the chromosome object with the highest fitness among the solution sets is selected. In the send() operation, information about the data forwarding sensor nodes is sent to the child sensor nodes. As the population has evolved over many generations, chromosome objects with lower fitness are eliminated from the solution sets. Otherwise, chromosomes with the higher fitness remain continuously.

4 Experimental environment

4.1 Tools and scenario

To evaluate the performance of the TARP, experiments have been performed on the TinyOS and TOSSIM tools [8]. The TinyOS is a component-based operating system for wireless sensor networks developed at Berkeley. The TOSSIM is a simulator of the TinyOS. In our experiments, we place 100 wireless sensor nodes uniformly as a grid structure and use the routing path of tree construction. To generate the data traffic, the surge program is used as the application of sensor nodes [8]. This program transfers the sampling data of a photo sensor to a sink node. The traffic congestion protocols are merged into the Surge application.

The scenarios are used to carry out more detailed experiments. The initial data transfer rate is 0.25 packets per 1 second. After 50 seconds have passed since the initial point, the events take place. When the events are detected, the data sampling rates in the terminal sensor nodes increase to get more information. The collected data are transferred to the sink node via the multiple hops of the sensor nodes located in the routing path. After 50 seconds, the data traffic in entire sensor networks increases with the higher data sensing rates. In these scenarios, traffic congestion appears in the relay sensor nodes and the surroundings of the sink node. To evaluate the performance of the TARP, we also measure the performances of the MINT and CODA methods. They were proposed for traffic congestion in wireless sensor networks. The MINT is the traditional routing protocol used in the TinyOS [8]. The CODA is a previous solution for congestion detection and avoidance in wireless sensor networks [6]. They use the same experimental conditions.

4.2 Performance metrics

For the purpose of the performance metrics, we use the effectiveness (η) and the fairness (Ø) of data transmission. Equation (3) shows the effectiveness of data transmission. It is achieved by dividing the transmission hops of packets (u) received in the sink node by the sum of the relaying packets and the packets created for itself. The effectiveness represents how effectively sensor nodes use the limited network bandwidth. Since the congested traffic causes a queue overflow in the sensor nodes, the packet losses increase during the packet relay process. As a result, low effectiveness has been achieved.

Equation (4) shows the fairness of data transmission in entire wireless sensor networks. The fairness is driven from the data transfer rates (r_i) of the sensor nodes. In our experiments, we measure the fairness of the sensor nodes of the same depth position in a routing tree. Wireless sensor nodes operate on limited battery energy and relay event data via multiple hops. The congested traffic to specific sensor nodes quickly exhausts their battery energy. Since a sensor node with an exhausted battery creates an energy hole in a routing path, the lifetime of the wireless sensor networks may be reduced. From the fairness metric, we can also evaluate the distribution of energy consumption in wireless sensor networks caused by data transmission.

$$\eta = \frac{\sum_{u \in U} \mathrm{hops}(u)}{\sum_{n \in N} (\mathrm{relay}(n) + \mathrm{create}(n))} \quad (3)$$

$$\phi = \frac{(\sum_{i=1}^{N} r_i)^2}{N \sum_{i=1}^{N} r_i^2} \quad (4)$$

- N: total number of sensor nodes
- r: data transfer rate
- U: set of packets received in the sink node

5 Performance evaluation

5.1 Effectiveness of data transmission

As mentioned in (3), the effectiveness is the ratio of the hops of the total packets received in the sink node to the number of packets transmitted from all sensor nodes. Low effectiveness means that even if sensor nodes transmit many data packets, the packet reception rate of the sink node is low. The lost packets in the middle of multiple hops are a major source of wasted energy in wireless sensor networks. Since the wireless sensor networks have worked on limited energy

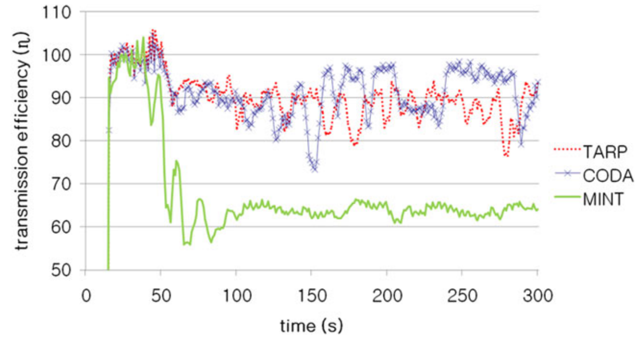


Fig. 8 Effectiveness

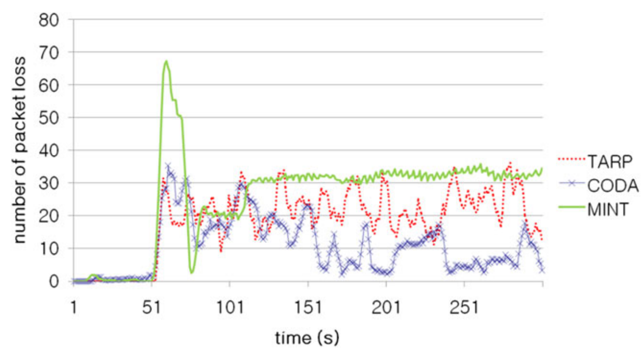


Fig. 9 Packet loss due to queue overflow

resources, the high effectiveness is an important requirement for the long lifetime of wireless sensor networks.

Figure 8 shows the variations in effectiveness in the TARP, CODA and MINT methods. Until 15 seconds, they have an initialization phase forming a routing path. According to the scenarios described in Sect. 4.1, an event takes place at 50 seconds so that the network traffic to collect the detailed information increases. However, even in the increased traffic, the TARP and the CODA sustain effectiveness at the 0.8 level. Otherwise, the MINT shows effectiveness at the 0.65 level.

After the event occurs, the MINT increases the sensing rates of the terminal sensor nodes to get more detailed information and begins to relay the sensing data. Due to this characteristic, the MINT generates queue overflows continuously because the specific sensor nodes in a routing path suffer from the network traffic. In Fig. 9, we can confirm the number of packet losses due to queue overflows. These queue overflows have the MINT to get the low effectiveness.

However, the CODA decreases the sensing rate of the terminal sensor nodes when the traffic congestion occurs. The queue overflow problem can be avoided by reducing the data sensing rate. As shown in Fig. 9, due to small packet losses by the decrease in queue overflows, the CODA can sustain the high effectiveness value when compared to the MINT. However, in the CODA, the sink node cannot achieve enough information to the events because the data sensing

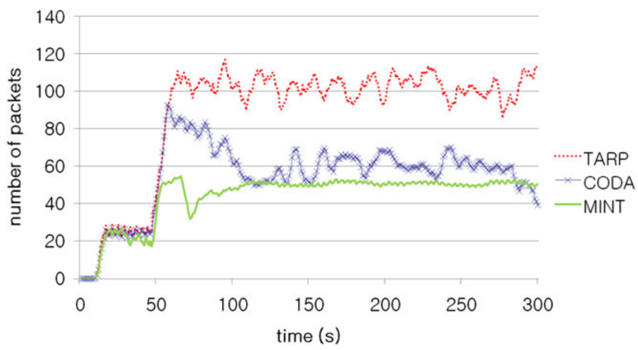


Fig. 10 Number of received packets in sink node

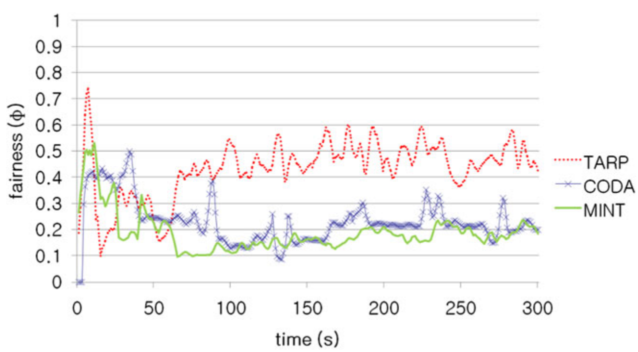


Fig. 11 Fairness

rate is reduced. Figure 10 shows the number of packets received in the sink node. When using the CODA, we can confirm that the sink node receives greatly reduced numbers of data packets after the event. Since insufficient sensing data make it difficult to interpret the event states, the complete role of wireless sensor networks cannot be achieved.

In the TARP, after the event occurs, the sensing rates from the terminal sensor nodes are increased to get the more accurate information. Since the data traffic rise with the increased data sensing rates, the traffic congestion could be generated in specific sensor nodes. However, since the TARP distributes the concentrated traffic into the available neighbor sensor nodes, the traffic congestion could be relieved.

As shown in Fig. 9, the TARP has low queue overflows when compared to the MINT. After the event is detected, the TARP makes the sink node receive enough sensing data to analyze the event states. From the result of Fig. 10, the TARP shows that the sink node receives more data packets when compared to the MINT and the CODA. Due to the high effectiveness, enough sensing data, and low queue overflow, we confirm that the TARP can support efficient resource usage and the reliable data transfer in wireless sensor networks.

5.2 Fairness of data transmission

In a routing tree, we measured the fairness of sensor nodes deployed in the upper level of the location where the events happened. The fairness of sensor nodes is driven from (4). Figure 11 shows the variations in the fairness among the TARP, CODA, and MINT methods. After an event occurs at 50 seconds, the network traffic increases to relay the packet data with the event information to the sink node.

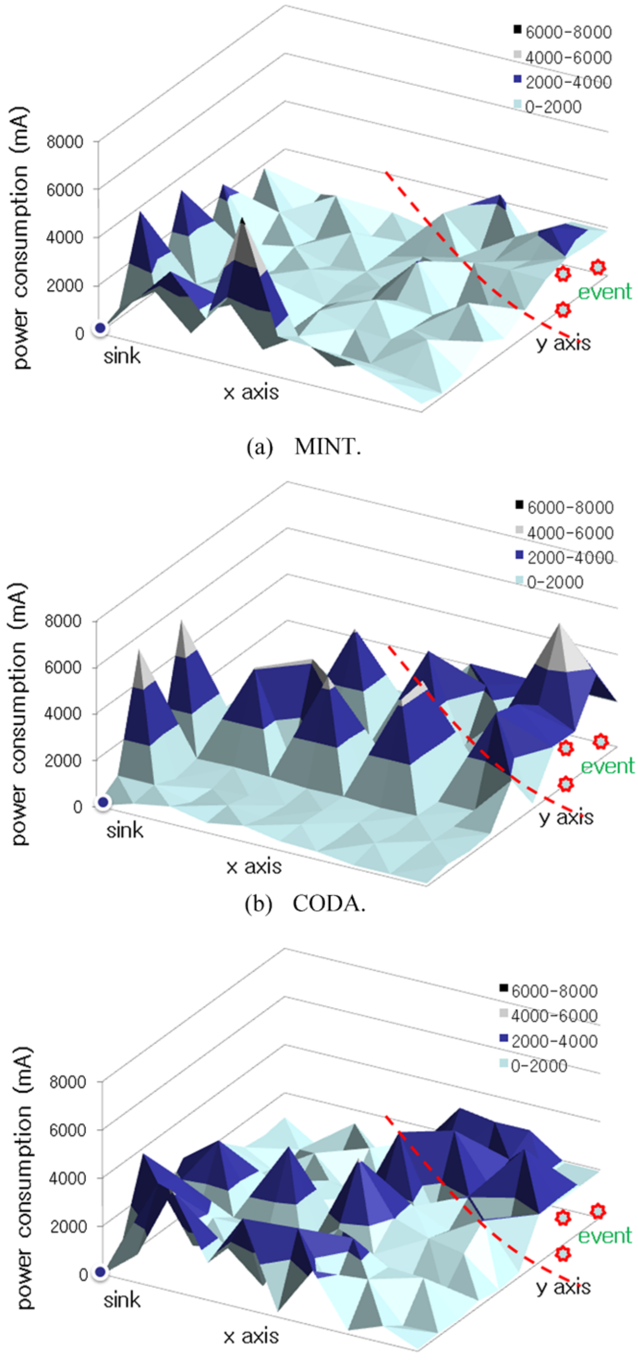
In the CODA and MINT, since only the specific sensor nodes treat the increased data traffic, they have fairness values of 0.23 and 0.19, which are relatively low. Otherwise, since the TARP distributes the data traffic to neighbor sensor nodes, this protocol represents a fairness of 0.34. However, although the TARP has a better fairness value when compared to the CODA and the MINT, the TARP's value is not high. The reason is that the TARP chose the data forwarding sensor nodes among the neighbor sensor nodes within 2 hops from the child sensor nodes of the sensor nodes suffering from traffic congestion. To get a higher fairness value, instead of the neighbor sensor nodes within 2 hops, the other sensor nodes located in the long distance can be chosen. However, this approach causes the wireless sensor networks to consume more battery energy because the data packets generated from the terminal sensor nodes should be relayed to the sink node via the designated long-distance path.

5.3 Power consumption

Figure 12 shows the distribution of energy consumed in all sensor nodes. As mentioned in Sect. 4.1, the sensor nodes are uniformly disseminated on the bottom square area. A sink node is deployed on the left vertex of the bottom square. The events are issued in the right area that the terminal sensor nodes are located. After the events take place, the power consumption increases according to the routing path along which the data packets are relayed. In the MINT, when compared to other methods, many sensor nodes consume 4000–6000 mA, which means that the energy consumption is disproportionate across entire wireless sensor networks.

Since the CODA regulates the data sensing rate of the terminal sensor nodes that detect the events, its power consumption is sustained at a low level. However, the reduced data sensing rates give rise to difficulties in accurate analysis of the events. Otherwise, the TARP consumes 2000–4000 mA even if the data sensing rates are increased. These energy consumptions are much smaller than those of the MINT. However, when compared to the CODA, the sensor nodes located outside the routing path consume more energy. The reason is that the TARP distributes the data traffic to the neighbor sensor nodes of the sensor nodes suffering from traffic congestion.

In the TARP, the sink node can take 2 times data packet reception rates when compared to the CODA. This effect



(a) MINT.

(b) CODA.

(c) TARP

Fig. 12 Power consumption

results in an increase in power consumption. However, the TARP can prevent extreme energy exhaustion in specific sensor nodes located in the routing path. This effect contributes to extending the lifetime of the wireless sensor networks worked on the multiple hops. In addition, even if traffic congestion occurs, since the TARP does not require the reduction of data sensing rate, the TARP can provides the sink node with enough information to analyze the state of events.

6 Conclusion and future work

Since wireless sensor networks have been deployed in isolated circumstances, saving battery energy is important for extending the lifetime of wireless sensor networks. After the events were detected, the amount of sensing data increased. These sensing data are transmitted to a sink sensor node via the routing path composed of multiple hops. In such an environment, the traffic congestion problem appeared in specific sensor nodes located in the chosen routing path. The traffic congestion not only caused data losses due to queue overflow but also reduced the effective lifetime of entire wireless sensor networks.

In this paper, the TARP was proposed to alleviate the traffic congestion problem in wireless sensor networks. It used the lightweight genetic algorithm to select the data forwarding sensor nodes. These sensor nodes took over the data traffic of the sensor nodes suffering from heavy network traffic. Based on the information of the neighbor sensor nodes, our lightweight genetic algorithm devised a chromosome structure for the sensor nodes and performed genetic operations such as crossover and mutation. To find the chromosome with optimal data forwarding sensor nodes, a fitness function was proposed based on the average and the standard deviation of the data transfer rates of all sensor nodes.

In experiments, we evaluated the TARP and two previous methods with the metrics of the effectiveness and the fairness. In addition, the distribution of consumed energy in wireless sensor networks was measured. The TARP showed as much as 15% higher effectiveness compared to the MINT and CODA methods. In the fairness, the TARP represented as 20%–24% higher than the other two methods. In addition, we confirmed that the MINT and CODA resulted in disproportionate energy consumption in wireless sensor networks. However, since the TARP could prevent the extreme energy exhaustion of specific sensor nodes located in the routing path, the TARP contributed to extending the lifetime of wireless sensor networks.

In our future work, we plan to design a new fitness function reflecting the remaining battery power together with data transfer rates. In addition, where wireless sensor networks have multiple sink nodes, a new algorithm to search an optimal sink node will be studied.

References

1. Akyildiz, I.-F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Comput. Netw. **38**, 393–422 (2002)

2. Polastre, J., Szewczyk, R., Sharp, C., Culler, D.: The mote revolution: low power wireless sensor network devices. In: Proceedings of Hot Chips 16: A Symposium on High Performance Chips, August 2004
3. Sankarasubramaniam, Y., Akan, O.B., Akyildiz, I.F.: ESRT: event-to-sink reliable transport for wireless sensor networks. In: Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing, June 2003, pp. 177–188 (2003)
4. Maimour, M., Pham, C., Amelot, J.: Load repartition for congestion control in multimedia wireless sensor networks with multipath routing. In: Proceedings of the 3rd International Symposium on Wireless Pervasive Computing (ISWPC'08), pp. 11–15 (2008)
5. Dietrich, I., Dressler, F.: On the lifetime of wireless sensor networks. ACM Trans. Sens. Netw. (TOSN) **5** (2009)
6. Wan, C., Eisenman, S., Campbell, A.: CODA: congestion detection and avoidance in sensor networks. In: Proceedings of the 1st ACM Conference on Embedded Networked Sensor Systems, November 2003, pp. 266–279 (2003)
7. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison–Wesley, Reading (1989)
8. http://tinyos.net

Chongmyung Park received his BS and MS degrees in Computer Engineering from Kangwon National University in 2005 and 2007, respectively. He is currently a PhD candidate in Computer Engineering at Kangwon National University and a visiting scholar at University of Minnesota Duluth. His research interests include wireless sensor network, parallel processing and embedded system.

Harksoo Kim is an assistant professor of computer and communications engineering at Kangwon National University. He received his BA in Computer Science from Konkuk University in 1996. He received his MS and PhD in Computer Science from Sogang University in 1998 and 2003, respectively. He visited the CIIR at the University of Massachusetts in Amherst as a research fellow in 2004. In 2005, he worked for Electronics and Telecommunications Research Institute (ETRI) as a senior researcher. His research interests include natural language processing, data mining, information retrieval, and question-answering.

Inbum Jung received his BS degree from Korea University, in 1985 and the MS and PhD degrees in Computer Science from KAIST, in 1994 and 2000, respectively. From 1984 to 1995, he was with Samsung Electronics Co. Ltd., Korea. He is currently a faculty member at Kangwon National University. His research interests include operating system, parallel processing, streaming media and wireless sensor network.