Energy-efficient Data Gathering Techniques Using Multiple Paths for Providing Resilience to Node Failures in Wireless Sensor Networks

Joungsik Kim and Dongkyun Kim*
Department of Computer Engineering, Kyungpook National University, Daegu, Korea
Email: {jskim, dkkim}@monet.knu.ac.kr

Abstract— **Multiple paths have been used to provide load-balancing of traffic and resilience to node failures in mobile ad-hoc networks or sensor networks. In addition, the energy expenditure can be distributed among nodes over the multiple paths. However, most existing multi-path routing protocols require their complex procedures such as local repair or periodic reconstruction of the multiple paths, during which much loss of packet is even experienced. In this paper, we therefore propose two versions of an Energy-efficient Data Gathering technique using Multiple paths (EDGM), which are more suitable for highly populated wireless sensor networks in terms of having more simple and resilient features than other existing techniques. One of them is to make use of the multiple shortest paths strictly. The other is to use a neighbor node's path towards the sink node at the expense of using much longer paths to provide more resilience to node failures. By simulation using ns-2 simulator, we prove that EDGM versions achieve two goals satisfactorily: energy-saving and resilience to node failure.**

Index Terms— **wireless sensor network, resilience, energy-saving, multiple paths, neighbor knowledge**

I. INTRODUCTION

The current advanced technologies of digital signal processing mechanism, sensing and wireless networking devices have allowed various sensor applications such as remote monitoring for an interested event to be used easily in the near future [1]. In the sensor networks, sensor nodes are generally scattered in an interested area to transmit events or data sensed at source nodes to a sink node required to process those data. The high cost of wiring sensor nodes requires the use of wireless links with short-range radio capability, which makes the nodes rely on multi-hop wireless forwarding services for the propagation of sensed data toward a sink node located far from source nodes.

In addition, the reliable transmission of sensed data over error-prone wireless links and fault-prone sensor nodes is crucial from source nodes to a sink node. Transport layer such as TCP (Transmission Control Protocol) can provide the reliability with much overhead for in-order and error-free delivery. However, network layer can also provide the reliability to an extent in order to reduce the burden of transport layer by using multiple paths which are easily obtained due to the mesh topology of wireless sensor networks. In this paper, we focus on the network-layer effort to provide the reliability through the use of multiple paths.

The multiple paths in a source-sink pair can be simply obtained as in much research effort [2] [3] [4] [5]. In addition to the multiple path acquisition technique, an energy-saving technique is essentially required in the wireless sensor networks because node failures due to the early energy depletion can make the network short-lived. In this sense, it is more important to make the best use of the multiple paths by considering the energy consumption at nodes.

In this paper, we therefore propose an energy-efficient data gathering technique using multiple paths, called EDGM (Energy-efficient Data Gathering with Multiple paths). EDGM has two goals: balanced energy consumption and resilience to node failures. For the purpose of providing the reliability and resilience at network layer, source nodes utilize multiple paths toward a sink node. However, when a node picks up a next-hop node toward the sink node, a node with the largest amount of residual energy is selected as the next-hop node in order to balance the energy consumption among all nodes. In addition, EDGM introduces two solutions to address a case where a node cannot participate in forwarding due to the energy depletion of its all next-hop nodes. The first solution is to find the nodes which are located at up-stream position of the node over the shortest paths and which have other paths towards the sink node through a back-tracking technique. It makes use of the multiple shortest paths strictly. The second one is to find the node's neighbor nodes which still have the paths towards the sink node (called EDGM with neighbor knowledge in this paper).

The rest of this paper is organized as follows. In Section 2, we describe some closely related work on

---

Based on ”An Energy-efficient Data Gathering Technique using Multiple Paths in Wireless Sensor Networks”, by D.K. Kim, J.S. Kim and K.H. Kim which appeared in the Proceedings of the IEEE Consumer Communications and Networking Conference 2006, Las Vegas, USA, January 2006. © 2006 IEEE.

* The corresponding author is Dongkyun Kim, dkkim@monet.knu.ac.kr.

multiple path routing. Section 3 presents the basic EDGM technique in detail, which is followed by the EDGM version with neighbor knowledge in Section 4. In Section 5, the performance evaluation is presented. Finally, some concluding remarks are given in Section 6.

II. RELATED WORK

Much research effort on multiple path routing protocols has been made for two main goals: load-balancing and fault-tolerance. For fault-tolerance feature, multiple path routing has been explored for reliable data delivery by sending multiple copies of the same data to multiple paths simultaneously, which provides a resilience to the failure of some paths. For load-balancing feature, traffic in a source-destination pair is split across multiple paths, which produces the overhead for deciding how to split the traffic across different paths.

In addition, much research work has been performed on multiple path routing [2] [3] as well as energy consumption of nodes. [2] has much overhead that network topology or data transmission rate should be known in advance for solving its linear programming problem to maximize the network lifetime. In [3], a path is selected in a probabilistic manner with the cost corresponding to each path maintained. For the path maintenance, it relies on infrequent local flooding, which lacks the resilience to reliability and node failure.

In [4], a sink node periodically performs the multiple-path construction from source nodes to the sink node. Thereafter, sensor nodes utilize the multiple paths for increasing the reliability and sharing energy depletion among all sensor nodes. Although it seems simple compared to other schemes, the recovery of node failure relies on only a periodic reconstruction of the multiple paths. In [5], a resilient multipath routing protocol was proposed using two paths, which requires local repair mechanism. However, utilizing multiple paths with over two paths can avoid the overhead of the local repair and achieve the energy-saving more satisfactorily.

III. EDGM : ENERGY-EFFICIENT DATA GATHERING WITH MULTIPLE PATHS

Our EDGM scheme consists of two phases: multiple-path construction phase and data forwarding phase. In those two phases, the energy saving effort is made.

A. Multiple Path Construction Mechanism

In this phase, sources-to-sink shortest multiple paths are periodically constructed. A sink node periodically floods **Multi_Path_Const** message into the network.

During the flooding of the message, each intermediate node creates its routing entries for the next-hop nodes over all the shortest reverse paths toward the sink node.

In case that a node attempts to obtain all multiple paths toward the sink node, instead of the shortest multiple paths, we suffer from the overhead of flooding **Multi_Path_Const** message and the complexity of avoiding a loop in the network. Therefore, all nodes including source nodes make the shortest multiple paths toward the sink node. Our procedure of acquiring the multiple shortest paths at each node is described in Algorithm 1.

Algorithm 1 Multiple Shortest Paths Construction Mechanism

```
$current\_hop\_cnt$ : hop-count of the shortest path until now
$recv\_hop\_cnt$ : hop-count of the received $Multi\_Path\_Const$ message
if (Multi_Path_Const Received) then
    if ($current\_hop\_cnt < recv\_hop\_cnt$) then
        drop the message;
    else if ($current\_hop\_cnt == recv\_hop\_cnt$) then
        insert the sender into routing table as a next-hop node;
        drop the message;
    else
        remove all the existing next-node entries in the table;
        insert the sender into routing table as a next-hop node;
        $current\_hop\_cnt = recv\_hop\_cnt$;
        increase the hop count of the message and rebroadcast it;
    end if
end if
```

From the energy consumption's view, the use of the shortest paths allows the total transmission power to be minimized because all sensor nodes use the fixed transmission range.

B. Data Forwarding Mechanism

However, we should also consider residual energy at nodes because only using the shortest path forces nodes with low residual energy over the path to exhaust their energy, so that they cannot participate in forwarding data any more. Therefore, the small number of nodes alive in the network results in reducing the availability of multiple paths and finally performance degradation.

To address the problem, when a node picks up a next-hop node, it should select a path with the next-hop node whose residual energy is the largest among next-hop nodes over its multiple paths toward the sink node. In other words, our EDGM tries to consider residual energy as well as to minimize the total transmission power. In particular, since each node performs the periodic exchange of **hello** messages to inform its neighbor nodes of its residual energy, the absence of the periodic message from a node's neighbor node means that the neighbor node lacks the residual energy so that it cannot participate in forwarding.

In order to take advantage of the availability of multiple paths and the resilient feature to a node failure, we propose a back-tracked forwarding technique in this paper. Suppose a node recognizes that all next-hop nodes over multiple paths lack their residual energy so that they cannot take part in forwarding (that is, there is no routing entry indicating that the residual energy of its next-hop node towards the sink node is greater than 0.).

In the case, the node will notify its previous node of its disability of forwarding by broadcasting a **BACK_TRACK** message. On receiving the **BACK_TRACK** message, the node using the sender of the message as its next-hop node in its routing table will exclude the next-hop node from its set of multiple paths.

If the back-tracking process reaches a source node and there exists no next-hop node of the source node, it can decide one of two choices. One of them is to wait for another periodic **Multi_Path_Const** message from its sink node. The other is to acquire the route from itself toward the sink node and use the route until the periodic **Multi_Path_Const** meets the node. In EDGM, we chose the latter approach for improving the packet delivery ratio. Our back-tracked forwarding mechanism is described in Algorithm 2.

Algorithm 2 Back-tracked Forwarding Mechanism

```
if (Data Received) then
  if (No available path to sink) then
    broadcast BACK_TRACK message;
  end if
end if
if (BACK_TRACK Received) then
  remove the sender of the message from the next-hop list;
  if (No available path to sink) then
    if (source node) then
      acquire a new path from itself to the sink using unicast
      routing protocol;
    else
      broadcast BACK_TRACK message;
    end if
  else
    drop the BACK_TRACK message;
  end if
end if
```

C. An Illustrative Example

We illustrate how our EDGM scheme works by using an example shown in Figure 1. The numbers around nodes depict the amount of residual energy. After one multiple shortest paths construction phase, the multiple shortest paths in a source-sink pair were obtained as shown in Figure 1. In this figure, the source node picks up $node\ g$ as its next-hop node because $node\ g$ has the largest amount of residual energy among its all next-hop nodes. Similarly, the sink node becomes reached through $node\ e$ to $node\ a$.

Suppose that $node\ a$ exhausted its residual energy. Finally, the absence of periodic **hello** messages allows $node\ b$ to recognize that all next-hop nodes of $node\ b$ toward the sink node exhaust their energy and there exists no available path to the sink node. Therefore, $node\ b$ broadcasts a **BACK_TRACK** message. On receiving the message, $node\ e$ also recognizes no available path to the sink node, which causes it to broadcast a **BACK_TRACK** message. However, since $nodes\ f$ and g have other available paths, they can avoid forwarding data to $node\ e$ and forward data to $nodes\ c$ and d, respectively.

In addition, suppose that $nodes\ c$ and d exhaust their energy. In this case, $nodes\ f$ and g broadcast their **BACK_TRACK** messages. Finally, since the source node has no available path to the sink node, it attempts to obtain a unicast path ($path\ k-l-m-n$ in the figure) and use the path until new multiple paths are set up.

D. Discussion

In EDGM, all nodes periodically attempt to obtain the multiple shortest paths from source nodes to a sink node as in [4]. We can easily acquire multiple shortest paths in dense sensor networks, which enables us to make the best use of the reliability and energy saving effect. In contrast, in case of sparse sensor networks, we cannot improve the performance due to lack of multiple shortest paths.

However, since the multiple shortest paths are always required, there exists some cases where nodes acquire the small number of shortest multiple paths (in the worst case, only one shortest path) although there are many other paths with more hops than the shortest path. In particular, the continual failures of nodes over time due to lack of energy cause the situation to be worse. Finally, the nodes over the shortest path can exhaust their energy and cannot participate in forwarding. In this case, we should wait for other **Multi_Path_Const** message from the sink node. Therefore, the period of the multiple path construction is critical to the performance, which [4] suffers from.

However, our EDGM approach requires a source node to acquire its path toward a sink node for transmitting data to the sink node until new multiple paths are obtained. For the purpose of acquiring a unicast path, the AODV (Ad-hoc On-Demand distance Vector) protocol [6] originally proposed for mobile ad hoc networks can be used temporarily.

IV. EDGM WITH NEIGHBOR KNOWLEDGE

In the basic EDGM scheme mentioned in previous sections, each node simply forwards packets to a node with the largest amount of residual energy over multiple shortest paths from a source node towards the sink node. It can achieve the goal of balancing energy consumption of the nodes. In particular, in case that all next-hop nodes over the multiple shortest paths towards the sink node cannot participate in forwarding packets due to lack of their energy, it relies on two schemes: (a) the **BACK_TRACK** message-based technique, and (b) the periodic reconstruction of multiple shortest paths. The **BACK_TRACK** message-based technique adheres to using the multiple shortest paths. In case that there are no shortest paths available, it should also rely on the periodic reconstructions in the end. If the resumption of packet transmissions relies on the periodic reconstruction of multiple shortest paths, the period affects the performance of the scheme because packets cannot be transmitted successfully to the sink node until the new multiple shortest paths are acquired.

In order to address the problem, an enhanced version of EDGM is introduced with neighbor knowledge. Each node exchanges **hello** message with its neighbors periodically. The **hello** message contains whether or not each neighbor node has a path towards the sink node. Suppose that a node cannot forward packet towards the

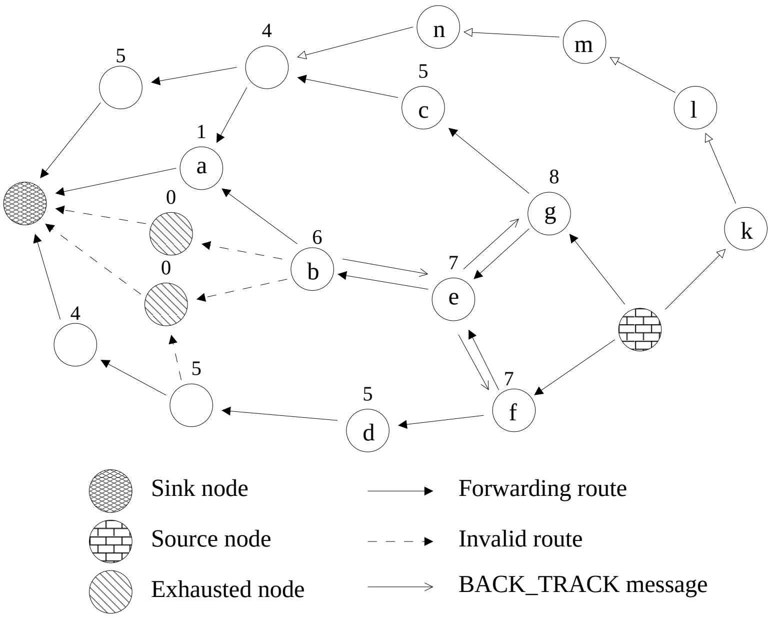
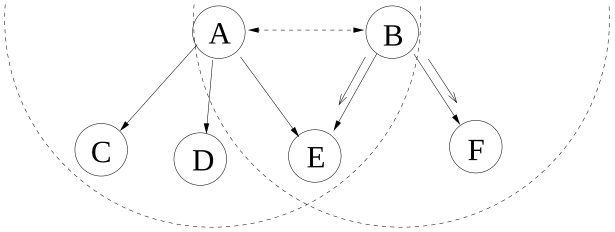
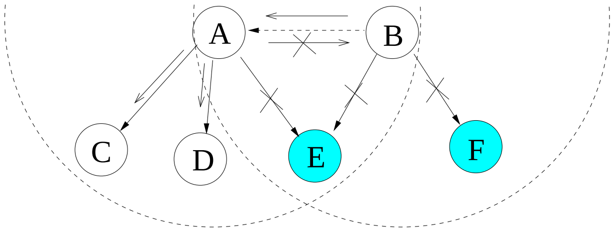


Figure 1. An Illustrative Example.

sink node because all next-hop nodes over the shortest multiple paths have died. If one of its neighbor nodes still has a path towards the sink node, the neighbor node is selected as a next-hop node even though the path towards the sink node increases by one-hop from the node's view. In other words, EDGM with neighbor knowledge uses a neighbor node's path towards the sink node at the expense of using much longer paths to provide more resilience to node failures. Therefore, even before new multiple shortest paths are acquired, the packet transmission will be continued with the help of neighbor nodes.



(a) Before node B lost all its multiple shortest paths.



(b) After node B lost all its multiple shortest paths.

Figure 2. EDGM with neighbor knowledge.

Algorithm 3 shows how the EDGM with neighbor knowledge works with back-tracked forwarding mechanism as a whole. After constructing multiple shortest paths, data forwarding phase is initiated. Each intermediate node forwards a received packet to one of its neighbor nodes with the largest residual energy over the shortest paths towards the sink node (in Figure 2 (a), when $node\ B$ had its next-hop nodes like $nodes\ E$ and F whose residual energy is enough to work over shortest paths between source and sink nodes, nodes E and F participated in forwarding in turn). If an intermediate node does not have its shortest paths available any longer ($node\ B$ in Figure 2 (b)), it selects one of its neighbor nodes ($node\ A$ in Figure 2 (b)) as its next-hop towards the sink because the **hello** message from the neighbor node indicates that the neighbor node has a path towards the sink node. The periodic **hello** message creates neighbor forwarding list consisting of the neighbor nodes that have those paths, if any. Therefore, the node with largest residual energy is selected for forwarding packets in order to provide balanced energy consumption of nodes. However, while using the path like $B - A - C(orD)$, when $nodes\ C$ and D, neighbor nodes of $node\ A$, cannot participate in forwarding due to lack of their residual energy, node A should notify $node\ B$ that the sink node cannot be reached via itself by broadcasting **NEIGHBOR_TRACK** message.

V. PERFORMANCE EVALUATION

We evaluated EDGM using ns-2 network simulator [8]. 200 nodes were initially positioned at random locations over 1000 m x 1000 m area. Each simulation runs for 300 seconds. For the simplicity of simulation in order to show relative performance improvement, we used IEEE 802.11 Wireless LAN of 2 Mbps as our MAC protocol, where every node had its limited transmission range of

Algorithm 3 Back-tracked Forwarding Mechanism with Neighbor Knowledge

```
if (Data Received) then
  if (No available path to sink) then
    if (No available neighbor with an available path to sink) then
      broadcast BACK_TRACK message;
    else
      if (Didn't send NEIGHBOR_TRACK message after Multiple Path Construction) then
        broadcast NEIGHBOR_TRACK message;
      end if
      send a data packet to the neighbor node;
    end if
  end if
end if
if (BACK_TRACK Received) then
  remove the sender of the message from the next-hop list and the neighbor forwarding list;
  if (No available path to sink) then
    if (No available neighbor with an available path to sink) then
      if (source node) then
        acquire a new path from itself to the sink using unicast routing protocol;
      else
        broadcast BACK_TRACK message;
      end if
    else
      if (Didn't send NEIGHBOR_TRACK message after Multiple Path Construction) then
        broadcast NEIGHBOR_TRACK message;
      end if
      send a data packet to the neighbor node;
    end if
  else
    drop the BACK_TRACK message;
  end if
end if
if (NEIGHBOR_TRACK Received) then
  remove the sender of the message from the neighbor forwarding list;
  if (No available neighbor with an available path to sink) then
    if (No available path to sink) then
      broadcast BACK_TRACK message;
    end if
  end if
end if
```

250 meters [1].

Simulations were performed by varying the number of source nodes with one sink node. A sink node was located at one corner of the area and other source nodes were located at the other corner. We used CBR (Constant Bit Rate) traffic sources with 40 byte-sized packet because sensor nodes generate their sensed data periodically (in the simulations, the period is set to 0.05 seconds to assume an environment where a sink node would like to collect data frequently). We set the initial amount of energy of each node to 10 Joules and the powers needed for transmitting and receiving the sensed data were set to 0.6 W and 0.3 W per second, respectively.

A. Performance Evaluation of EDGM without neighbor knowledge

Here, EDGM without neighbor knowledge is evaluated in order to investigate its performance improvement compared to the case using a unicast path from each source node to a sink node, the AODV unicast routing protocol was used in this simulation because it is usually used in mobile ad hoc networks or sensor networks. In particular, ZigBee-based sensor network utilizes AODV for its network routing protocol [7]. To acquire each unicast path between source and sink nodes, AODV floods a route request (RREQ) and it needs a unicasted route reply (RREP) to enable each forwarding node to select one next-hop node toward the sink node. Afterwards, the acquired path is used until it is broken. Therefore, it needs to recover the broken path, which takes a long time. In this sense, EDGM improves the performance in terms of more resilience to a node or path failure, which results in obtaining more throughput than the other (see Figure 3). For throughput comparison, we measured the total number of packets received at the sink node from all sources. Particularly, the higher the number of source nodes is, the more the throughput improvement is obtained.

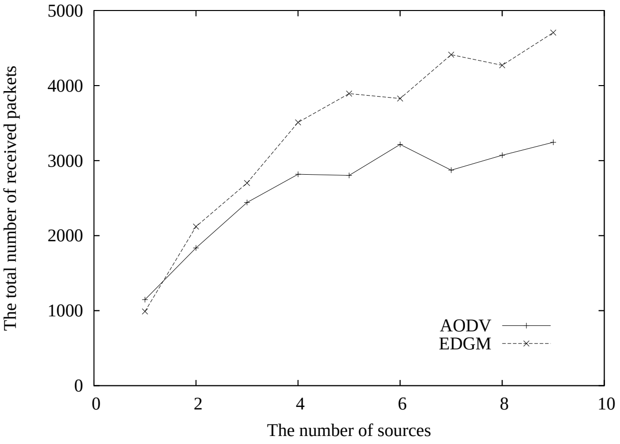


Figure 3. Throughput Comparison with AODV Unicast Paths.

In addition, we measured the time when we experienced the first occurrence of node failure due to lack of energy. Since the use of AODV needs more energy consumption of nodes over an acquired path until the path fails, the balanced energy consumption cannot be expected. However, EDGM allows the energy dissipation to be distributed evenly among nodes. Regardless of the number of sources, EDGM has better life times of nodes than AODV. In case that the number of source nodes is high, much contention and interference make the life times to be low because they cause more energy to be consumed (see Figure 4).

Next, we executed the performance comparison according to the way that each forwarding node selects its next-hop node toward the sink node. We compared EDGM with

---

[1] It is evident that many other low-rate MAC protocols suitable for sensor networks such as IEEE 802.15.4 can be applied, which is our future investigation.

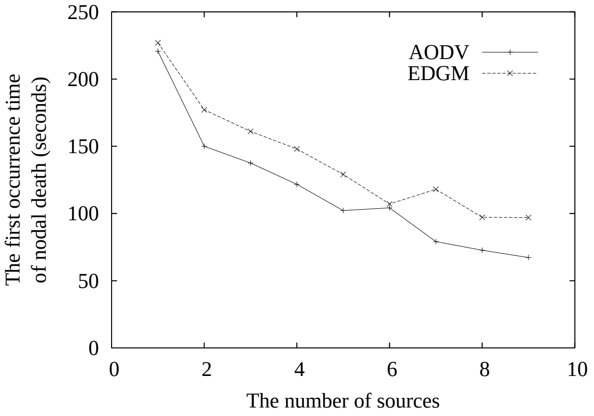


Figure 4. Nodal Life-time Comparison with AODV Unicast Paths.

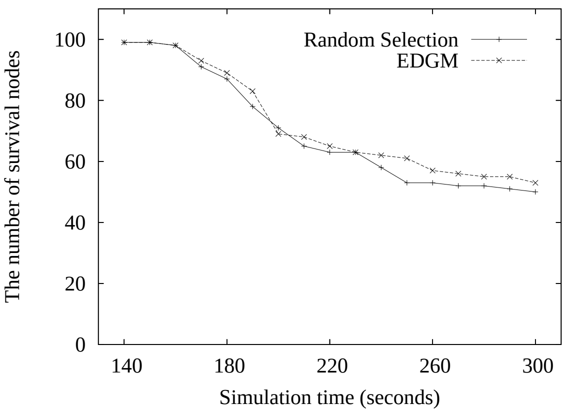


Figure 6. Nodal Life-time Comparison with Random Selection Technique (Sparse Network).

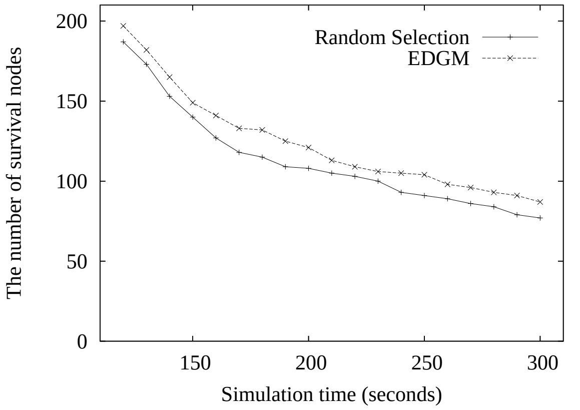


Figure 5. Nodal Life-time Comparison with Random Selection Technique (Dense Network).

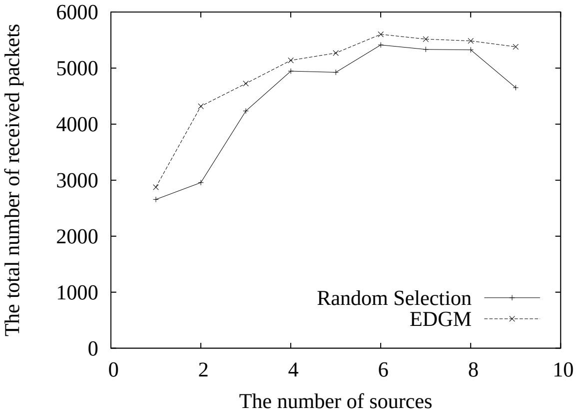


Figure 7. Throughput Comparison with Random Selection Technique (with 200 nodes).

the random selection technique where the next-hop node is randomly selected among next-hop nodes possible over multiple paths without considering the residual energy of nodes.

In case that the energy status of nodes is not considered, the energy consumption is not balanced among nodes, which results in causing nodes not to participate in forwarding further due to lack of residual energy. We measured the number of survival nodes over simulation time according to the density of nodes. In a dense network whose number of nodes is 200, EDGM obtained more survival nodes than the random selection technique due to the abundance of the multiple shortest paths (see Figure 5). In a sparse network with 100 nodes, we obtained a little improvement due to the small number of the shortest multiple paths (see Figure 6). In sequel, EDGM allowed more nodes to survive in the network due to its consideration of residual energy.

Finally, although both of approaches select one of next-hop nodes over candidate multiple paths, the random selection case has a risk to forward data to next-hop nodes which cannot participate in forwarding due to lack of energy, which results in much loss of packets. Therefore, EDGM still produced better throughput than the random selection technique (see Figure 7).

B. Performance Evaluation of EDGM with neighbor knowledge

Here, EDGM version with neighbor knowledge is compared with EDGM without neighbor knowledge which relies on the BACK_TRACK message-based technique in terms of two metrics: throughput and delivery ratio. Delivery ratio is defined as the ratio of the number of packets received at the sink node over the total number of packets transmitted by all source nodes. The same simulation environment with the same parameters that the previous simulations took are used in this simulation. We evaluated performance in two cases of node population, namely a sparse network (100 nodes) and a dense network (200 nodes). In each simulation, we varied the number of source from 1 to 9.

Figure 8 shows throughput comparison according to the number of source nodes when the number of nodes is 100 (that is, in a sparse network). Particularly, in case that the number of source nodes is small, the sparse network could not make many multiple shortest paths, which resulted in EDGM without neighbor knowledge experiencing more path failures. However, EDGM with neighbor knowledge could utilize more or less longer paths for the purpose of continuing packet transmissions to the sink node even if all its shortest paths are not available. In the scheme, a node not located over the multiple shortest path will participate in forwarding if its neighbor node still has paths towards the sink node. It results in more packets transmitted before new multiple shortest paths are acquired than in EDGM without neighbor knowledge. Hence, EDGM with neighbor knowledge has better throughput and delivery ratio regardless of node density. In terms of delivery ratio, EDGM with neighbor knowledge produced better performance than the other due to the same reason (see Figure 9).

As shown in Figure 10 and 11, in case that the number of nodes is 200 (that is, a dense network), more multiple shortest paths are expected to be available than in a sparse network. Even though we did not observe significant performance difference between two schemes, EGDM with neighbor knowledge could still show better performance due to its ability to use the detour paths until new multiple paths are acquired. Consequently, EDGM with neighbor knowledge is more suitable for providing resilience to node failures.

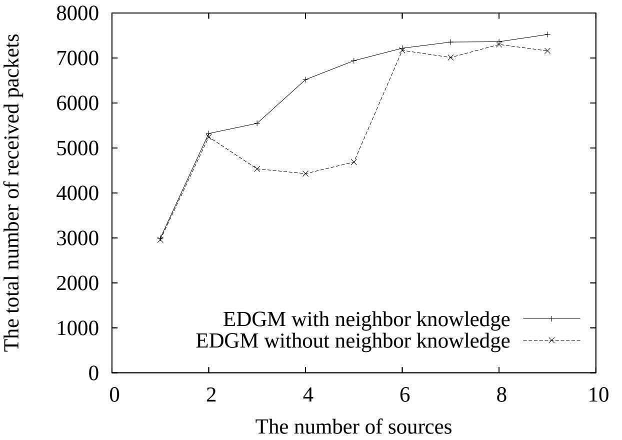


Figure 8. Throughput Comparison with Neighbor Knowledge Technique (Sparse Network).

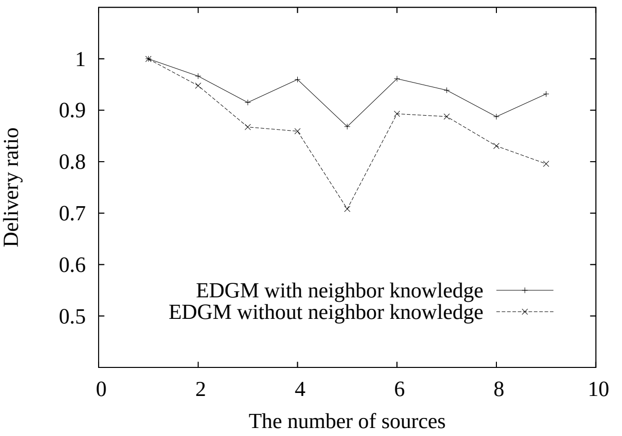


Figure 9. Delivery Ratio Comparison with Neighbor Knowledge Technique (Sparse Network).

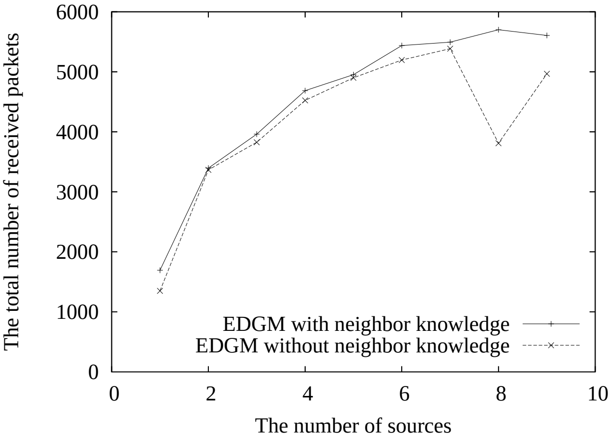


Figure 10. Throughput Comparison with Neighbor Knowledge Technique (Dense Network).

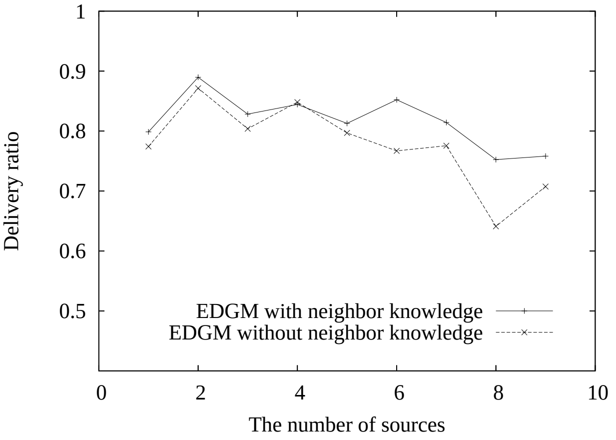


Figure 11. Delivery Ratio Comparison with Neighbor Knowledge Technique (Dense Network).

VI. CONCLUSION

We proposed two versions of EDGM (an Energy-efficient Data Gathering technique using Multiple paths) for wireless sensor networks. In addition to resilience to node failures and minimizing the total transmission power through the multiple shortest paths, in order to avoid overusing the energy of nodes over a single path and extend the life times of nodes, the forwarding nodes over the multiple paths and source nodes select the next-hop with the largest residual energy as next-hop node toward a sink node. Particularly, in case that a node cannot participate in forwarding any more because all its next-hop nodes cannot forward any data due to lack of residual energy, it allows other nodes to avoid selecting it as the next-hop node by broadcasting **BACK_TRACK** message or it selects one of its neighbor nodes as a next-hop node by utilizing neighbor knowledge. The former is to make use of the multiple shortest paths strictly, while the latter is to use its neighbor node's path towards the sink node at the expense of using much longer paths.

By using ns-2 simulator, EDGM showed better throughput and energy-saving performance than two approaches without EDGM capability, particularly in highly populated networks; (a) the data propagation relies on their unicast paths from each source to the sink node and (b) each forwarding node selects its next-hop node without considering residual energy of nodes. Consequently, our EDGM protocols are suitable for providing resilience to node failures as well as providing balanced energy consumption among nodes.

ACKNOWLEDGMENT

This work was supported financially by Winitech and Daegu Metropolitan City.

REFERENCES

[1] I.F.Akyildiz, W. Su, Y. Sankarasubramanian and E. Cayirci, “A survey on sensor networks,” *IEEE Communications Magazine*, August 2002.

[2] J.-H. Chang and L. Tassiulas, “Maximum Lifetime Routing in Wireless Sensor Networks,” In *Proc. of Adv. Telecommun. and Info. Distrib. Research Prog.*, College Park, MD, Mar. 2000.

[3] C. Rahul and J. Rabaey, “Energy Aware Routing for Low Energy Ad Hoc Sensor Networks,” In *Proc. of IEEE WCNC*, Orlando, FL, 2002.

[4] X.Hong, M.Gerla, H.Wang and L.Clare, “Load balanced, energy-aware communications for Mars sensor networks,” In *Proc. of IEEE Aerospace Conference*, March 2002.

[5] K.H.Kim and et. al., *A Resilient Multipath Routing Protocol for Wireless Sensor Networks*, ICN 2005, Lecture Notes in Computer Science, vol. 3421, 2005.

[6] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, “Ad-hoc On-Demand Distance Vector (AODV) Routing”, IETF RFC 3561.

[7] ZigBee Alliance Document 053474r06, Version 1.0: ZigBee Specification, December 14th, 2004.

[8] K.Fall and K.Varadhan, “ns notes and documents,” The VINT Project, UC Berkeley, LBL, USC/ISI. and Xerox PARC.

Joungsik Kim is now in MS course at the Department of Computer Engineering, Kyungpook National University, Daegu, Korea. He received the BS degree at Kyungpook National University in 2005. His current research interests include MANET, Sensor Network, and WLAN.

Dongkyun Kim is a professor with the Department of Computer Engineering, Kyungpook National University, Daegu, Korea. He received the BS degree at Kyungpook National University. He also obtained the MS and Ph.D degrees at Seoul National University, Korea. He was a visiting researcher at Georgia Institute of Technology. He also performed a post-doctorate program at University of California Santa Cruz. He has been a TPC member of several IEEE conferences. He received the best paper award from the Korean Federation of Science and Technology Societies, 2002. His research interest is ad-hoc network, sensor network and wireless LAN, etc.