A Dynamic Multiagent-Based Local Update Strategy for Mobile Sinks in Wireless Sensor Networks

Jinkeun Yu[1], Euihoon Jeong[1], Gwangil Jeon[1], Dae-Young Seo[1],
and Kwangjin Park[2,*]

[1] Department of Computer Engineering
Korea Polytechnic University, 429-703, Korea
`{yjk760,ehjeong,gijeon,seody}@kpu.ac.kr`
[2] Division of Electrical Electronic and Information Engineering
Wonkwang University, Iksan, 570-749, Korea
`kjpark@wku.ac.kr`

Abstract. Recently, several geographic routing strategies considering a mobile sink node have been proposed for wireless sensor networks. A mobile sink node should frequently update its location information in source nodes for successfully receiving data from the latter. However, frequent location updates from the mobile sink node may result in the consumption of too many network resources. In this paper, we propose an efficient multiagent-based location update strategy for a mobile sink node in wireless sensor networks. Agent nodes that are located on the path between the source node and the mobile sink node contain the location information of the mobile sink node. In addition, the agent nodes are changed dynamically to maintain the approximate shortest path between the source node and the mobile sink nodes. We analyze the performance of the proposed scheme by performing simulation using Qualnet 5.0.

Keywords: Wireless Sensor Networks, Mobile Sink, Location Management, Multiagent.

1 Introduction

Wireless sensor networks have been researched extensively and applied in a number of fields such as battlefield monitoring, residence monitoring, traffic congestion monitoring, and security. Sensor nodes are small and simple and use limited memories and batteries; hence, it is necessary to design a protocol for minimizing unnecessary transmission in wireless sensor networks [1]. A number of researches have been performed on networks with a fixed sensor node and a sink node, and recently, studies have been carried out on networks with mobile sink nodes as well. A mobile sink node is constantly in motion; therefore, changing the global network topology is inevitable. Changing the global network topology results in excessive energy consumption in large-scale sensor networks, and hence, geographic routing is frequently used [2-3].

---

[*] Corresponding author.

B. Murgante et al. (Eds.): ICCSA 2011, Part IV, LNCS 6785, pp. 185–196, 2011.

Geographic routing uses the location information of each node to transmit a data packet, and the location information of neighbor nodes can be confirmed through a beacon message. When transmitting a data packet, the source node includes the location information of the sink node in the packet. In addition, each intermediate node uses the location information of the sink node of a data packet to forward the packet to the next closest node. Since a mobile sink node is constantly in motion, its location information must be repeatedly updated in the source node [2-4]. Further, since a number of overheads are created during the path when updating the location information of a sink node, energy consumption increases. Therefore, it is necessary to come up with a method for effectively updating the location information of a mobile sink node. Some of these methods are grid-based methods such as TTDD (two-tier data dissemination) [5], local flooding methods such as ALURP (Adaptive Location Updates Routing Protocol) [6], and Elastic routing [7]. TTDD is a method in which a grid is used to transmit data. A mobile sink node within a local grid cell performs flooding of query messages to transmit them to a dissemination node, which informs the source node of the sink's location. However, a number of overheads occur when the size of a grid cell increases. In ALURP, the sink's location information is updated through flooding within an adaptive area. A source node transmits a data packet to a virtual center, and when this data packet encounters a node within the adaptive area, it receives the latest location information about the sink for transmission. Elastic routing involves updating a sink's location information along the same path but in the direction opposite to that of data transmission. In previous studies, the number of overheads in the location update of a mobile sink node was effectively decreased but not to a significant extent. Therefore, in this paper, we plan to decrease the number of overheads required when updating the location information of a sink node, by setting up an agent between a source node and a mobile sink node using the Elastic routing method.

In this paper, Dynamic MultiAgent-based Local Update Protocol (DMALUP) supporting a mobile sink is proposed. DMALUP is designed to set up one agent per constant hop and transmit data through this agent. Agents can be changed depending on the sink's location so as to support the approximate shortest path. Agent changing occurs when a value greater than the constant critical value is assigned to the straight line connecting the source node and the sink node. At this stage, when the node preceding an agent node has a value greater than the critical value, detection of new agents between the mobile sink nodes is initiated. Since DMALUP transmits the location information of a mobile sink node to the last agent, the number of overheads created by an update decrease. Thus, DMALUP is considered an efficient location information update process. The proposed method is capable of reducing the cost associated when updating a sink's location information and helps identify the approximate shortest path as the agent's location is changed.

The rest of this paper is organized as follows. Relevant past researches are introduced in section 2, and DMALUP is described in detail in section 3. In section 4, performance evaluation using the Qualnet 5.0 simulator is described, and in section 5, conclusions are presented.

2 Related Work

Some protocols propose a method for decreasing the number of overheads required for updating the location information of a mobile sink in wireless sensor networks. Initially, TTDD [5] uses a grid-based routing protocol to support the supply of queries and data to a mobile sink. When an event interested by a source node occurs, a global-grid structure is created in the overall network on the basis of the source node, and a dissemination node is assigned as the intersection point of the grid. A sink node performs flooding of query messages within a grid cell, and a close dissemination node uses the grid to transmit information about the location of the sink to a source node. The data are transmitted by the source node via the same path used to transmit the query messages, but in the reverse direction. TTDD only performs flooding of query messages within a local cell. It is an efficient method, but when the cell size increases, a number of overheads are created because of the aforementioned flooding. In addition, a number of overheads are created during the formation and maintenance of a global grid.

ALURP [6] sets the location of the initial mobile sink node as the virtual center, establishes a purpose boundary, and transmits the location of the virtual center to a source node. A source node transmits data to the virtual center, and a sink node performs flooding of its own location information to an adaptive area within the purpose boundary. Then, when a data packet transmitted by a source node arrives at a node within the adaptive area, the node uses the location information of the updated mobile sink node to transmit data. If a sink node moves out of the purpose boundary, the initial process is repeated. Since ALURP updates the location information of a mobile sink within the adaptive area, the overheads required for transmitting the location information of a mobile sink can be reduced. However, when the adaptive area enlarges, a number of overheads are created.

In Elastic routing [7], a source node uses greed forward to transmit data to a mobile sink node, and the updated location information of this node is transmitted in the reverse direction along the same path used for data transmission. Data transmission in this case is very efficient since the data are forwarded to the location of the new mobile sink node when the location information is encountered in the data transmission path. The service is executed in the order C-B-A, as shown in Fig. 1(a). However, when the sink node approaches node B, as shown in Fig. 1(b), the service is executed in the order C-B-Sink, and when the sink node escapes A’s transmission boundary, as shown in Fig. 1(c), the sink node greed-forwards location information to A. Then, a register stores sink’s new location information and transmits it to a source node in the order A-B-C. In Fig. 1(d–f), when the sink is reset to a new location, a data packet is greed-forwarded to the sink’s new route. Elastic routing always maintains the shortest route to reduce data transmission delay. However, since Elastic routing transmits the sink node’s location information to a source node every time the mobile sink node moves, a number of overheads are created. If the number of paths in the data transmission route is increased, more overheads are created.

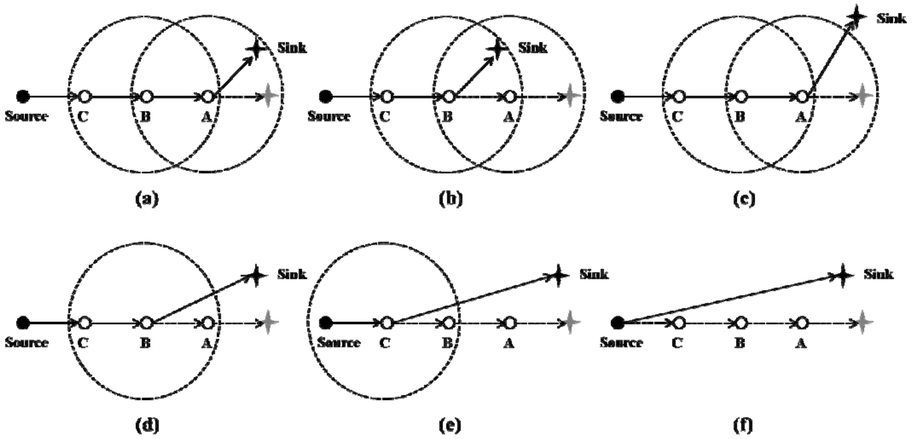

Fig. 1. Location propagation of a mobile sink

3 The Proposed Scheme

In this section, the proposed DMALUP is described in detail. DMALUP works under the following basic network assumptions:

1. The common node and the source node are fixed, and the mobile sink node is free to move very slowly.
2. All the nodes obtain their own location information using the GPS (Global Positioning System) or other location systems. They also obtain the location information of the neighbor nodes through a beacon message.
3. Initially, a source node is aware of the location of a sink node.
4. All the nodes are closely spaced.
5. Designing is executed under the assumption that there is no void area.

DMALUP sets up one static agent per constant hop and uses it to transmit data in order to decrease the number of control messages in the location information service of a constantly moving sink, while effectively transmitting a data packet. The number of control messages can be reduced since the location information of the sink is updated in the last agent.

Now, the path for setting an agent and transmitting data and the strategy for maintaining an agent and updating the location information of a sink node are presented.

3.1 Agents Setting and Data Packet Transmission

DMALUP uses agents to transmit data packets, and these agents assign an agent per constant hop interval between the transmitting source node and the sink node. Then, each agent memorizes the location information of the next agent and transmits the data packet along with this information to the next agent. The last agent then transmits the data packet to the sink node.

The procedures for setting up an agent between the source node and sink node are as follows.

1. A source node, as shown in Fig. 2(a), includes its own ID and location information in the agent list and transmits the sink ID, location information of the sink, hop intervals, and an agent list in the direction of the mobile sink node by using the greed-forward method.
2. The intermediate nodes that receive the message increase the hop and check if it counts as a hop interval. If it does not count as a hop interval, the message is greed-forwarded toward a sink node. If it is counted as a hop interval, the agent list is registered. Then, it adds its own ID and location information to the agent list and greed-forwards the message to a sink node.
3. When a sink node receives the message, it registers the agent list and uses the last agent, as shown in Fig. 2(b), to include its own ID and location information in the next ID and location, respectively, for a transmission.
4. An agent registers the next ID and location information received from the previous agent or sink node and updates its own ID and location information for transmission to the previous agent.
5. Lastly, when a source node receives a message, as shown in Fig. 2(b), it is registered in the next ID and location, thus ending the agent-setting course.

Fig. 2. Control message structure

When a data packet is transmitted, the location information of the next agent is included in the packet being transmitted. Data transmission between agents involves greed forwarding. Lastly, an agent uses the latest location information of a sink node to transmit data to it. If a particular agent lacks energy or does not perform its role, the surrounding nodes act on its behalf.

3.2 Sink Location Propagation and Agent Change

In DMALUP, since the location information of a sink node is updated not in a source node but in the last agent, the number of overheads can be reduced. When a data packet arrives at the final agent, the latest location of a mobile sink node can be confirmed. Such an agent can be dynamically changed. An agent changes in the following three cases: 1) when a sink node grows apart from the last agent, 2) when the sink node is closer to the source node than to the last agent node, and 3) when the agents grow apart from a straight line between the source node and the sink node.

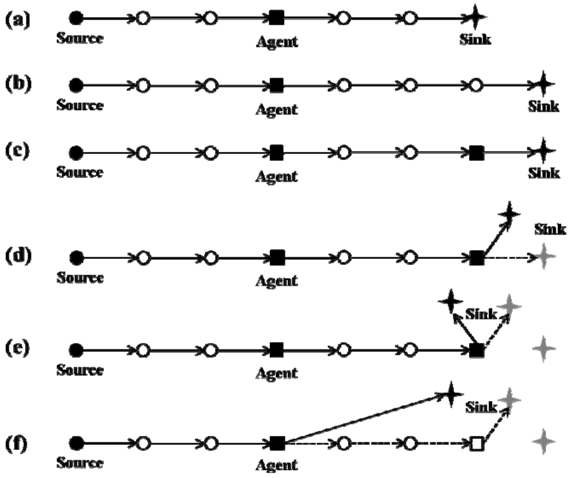

Fig. 3. When the sink node grows apart from the last agent node and the sink node is closer to the source node than to the agent

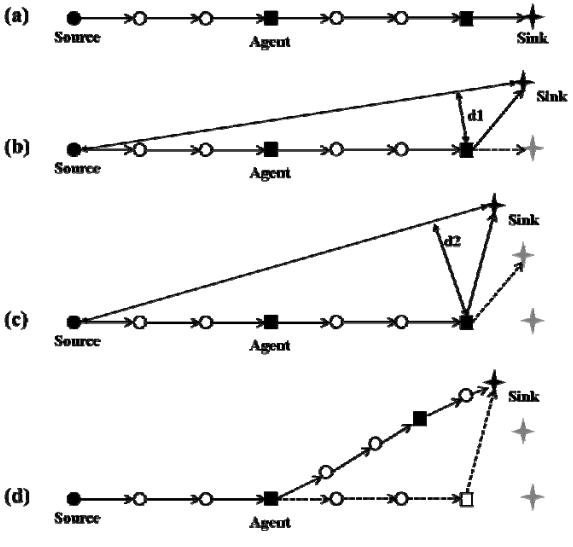

Fig. 4. When the agent node grows apart from a straight line connecting the source node and the sink node

In the initial case, the sink, as shown in Fig. 3(a), provides a service to an agent assigned to every three hops if the sink node grows apart from the last agent and the number of hops increases, as shown in Fig. 3(b). Then, the sink node transmits an agent update message to the final agent. As shown in Fig. 3(c), the last agent sets a new agent before sink node.

In the second case, the sink node is closer to the source node than to the last agent, and the node provides a service, as shown in Fig. 3(d). When the sink node is closer to the source node than to the last agent, as shown in Fig. 3(e), it transmits an agent update message to the previous agent, thus forming a new route from the previous agent.

The third case is when the agent grows apart from the straight line connecting the source node and the sink node. In this case, the purpose is to maintain the approximate shortest path by positioning an agent close to this straight line, even though this is not the shortest complete path. If the agent is configured and data are transmitted, as shown in Fig. 4(a), the sink node moves as shown in Fig. 4(b). In this case, the sink node on the path between the current location and the location of the source node does not change the agent if the distance to each agent, d1, is smaller than the threshold. If d2 is greater than the threshold (Fig. 4(c)), an update message is transmitted to the previous agent. At this stage, the message is transmitted in the direction opposite to that of data transmission. The agent receiving this message forwards it through a path for creating a new agent in the direction of the sink node.

4 Performance Evaluations

We use the Qualnet 5.0 network simulator [8] to evaluate the performance of the proposed DMALUP compared with Elastic routing. Then, we evaluate the performance depending on the maximum distance between the agents and the straight line connecting the source node and the sink node in DMALUP. The system parameters used in this simulation are presented in Table 1. For the sensor node, as shown in Fig. 6, 250 nodes are randomly distributed in a network of area 250 *m* × 250 *m*. The radio propagation range is set to 25 *m*. The IEEE 802.11b radio is used, and MicaZ is used as the energy model. The simulation is performed for 503 seconds. Three seconds of the simulation time is used to set the initial neighbor nodes.

As shows by the arrows in Fig. 5, the routes to the sink node are set to allow diagonal movement. This is to allow for frequent change of the agents. All the nodes transmitted a beacon message every second, CBR data at the rate of 1 *packet/s*, and a total of 500 data packets. Location information update for the mobile sink node is created every time this node moved 1 *m* away from its previous location.

Table 1. System parameters

Network Area	250*m* × 250*m*
Energy Model	MicaZ
Radio Type	IEEE 802.11b Radio
Size of Data Packet	512bytes
Radio Propagation Range	25*m*
Simulation Time	503*s*

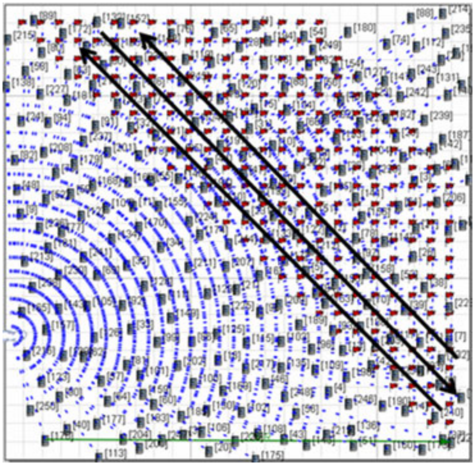

Fig. 5. Placement of nodes in the simulation environment

4.1 Simulation Results for Different Moving Velocities

We evaluate the performance of DMALUP and Elastic routing. The moving velocity of the mobile sink node is changed from 1 *m/s* to 10 *m/s*. DMALUP set an agent every three hops along the transmitting route between the source node and the sink node and the maximum distance of an agent from the straight line connecting the source node and the sink node is set in the range 30 *m*. The data transmission rate is found to be 100%.

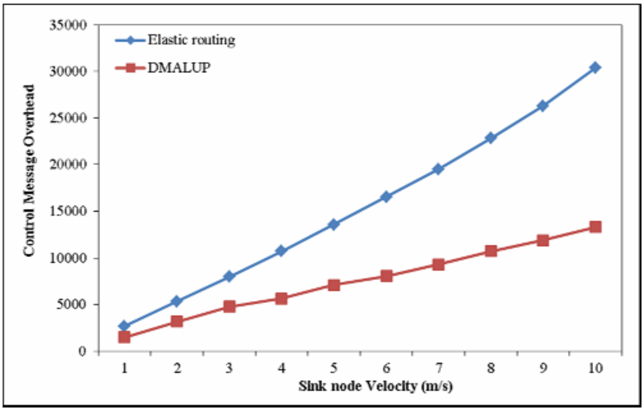

Fig. 6. Comparison of control message overhead

Fig. 6 shows the number of control message overheads in Elastic routing and DMALUP depending on the moving velocity of the mobile sink node. When the moving velocity is 1 *m/s*, the number of control message overheads in Elastic routing and DMALUP is 2680 and 1477, respectively. When the moving velocity is 5 *m/s*, the number of control message overheads in Elastic routing and DMALUP changed to 13570 and 7086, respectively. When the moving velocity is increased to 10 *m/s*, there are 30385 and 13299 control message overheads in Elastic routing and DMALUP, respectively. On an average, there is a 48% decrease in the overhead in the case of DMALUP. It is found that the location information of the mobile sink node is updated more efficiently in the case of DMALUP.

As shown in Fig. 7, the data transmission delays in the case of DMALUP and Elastic routing are approximately the same. In Elastic routing, the location information of the sink node is transmitted via the data transmission path but in the reverse direction, and message collision causes transmission delay. In DMALUP, the location information of the sink node is transmitted up to the last agent node, thereby reducing message collision. In Elastic routing, the transmission delay shows a zigzag pattern because a greater number of message collisions occur when the moving velocity of the sink node is an even number than when the moving velocity is an odd number, thus causing an increase in the data transmission delay.

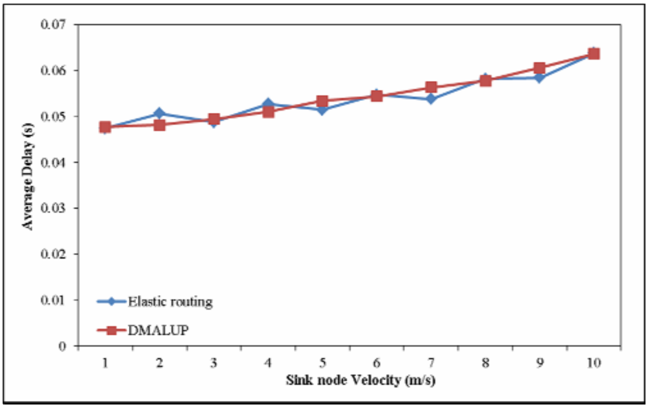

Fig. 7. Comparison of average delay

Fig. 8 shows the energy consume during a message transmission. The energy consume at this stage of transmission includes that consume during the transmission of the beacon message, data messages, and control messages. The overall energy consume during the beacon message transmission is approximately 17.67 *mJ*. When the cost associate with the transmission of a beacon message is excluded, the energy consumption in Elastic routing and DMALUP is 3,813 *mJ* and 3.70 *mJ*, respectively, for a moving velocity of 1 *m/s*. When the moving velocity is 5 *m/s*, the energy consume in Elastic routing and DMALUP is 6.52 *mJ* and 5.3 *mJ*, respectively. When the moving velocity is 10 *m/s*, the energy consumption becomes 11.10 *mJ* and 7.51 *mJ* in Elastic routing and DMALUP, respectively. Thus, we can see that the energy consumption in DMALUP is 2%, 18%, and 32% lower than that in elastic routing for moving speeds of 1 m/s, 5 m/s, and 10 m/s, respectively. No significant difference in energy consumption is seen when the moving velocity of the mobile sink node is low.

However, a significant difference is observed as the moving velocity increases. The reason for the difference in the results depicted in Fig. 6 and 8 is the overhead associated with agent maintenance in DMALUP and the fact that the data transmission route for DMALUP is slightly longer than that for elastic routing.

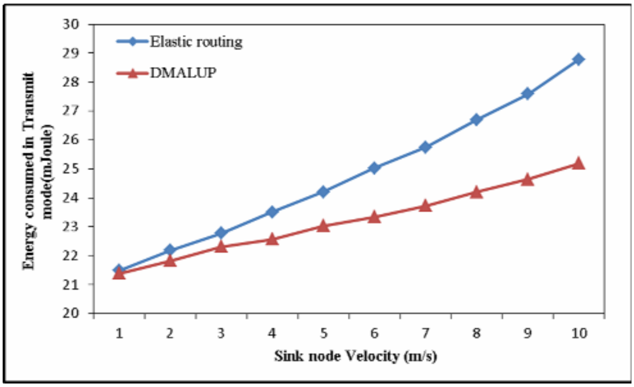

Fig. 8. Comparison of energy consumption during transmission

4.2 Simulation Results Depending on the Maximum Distance of the Agents from Straight Line Connecting the Source Node and the Sink Node

For DMALUP, the maximum distance of an agent from the straight line connecting the source node and the sink node is set in the range 10 *m* to 60 *m*. As shown in Fig. 9, the number of control message overheads is 13769 when the maximum distance is set to 10 m. This number decreases as the distance increases: 6332 at 40 *m*, 6350 at 50 *m*, and 6399 at 60 *m*. This is because the distance to the agent to be updated increases when the agent is updated. As shown in Fig. 10, the reason for the increase in the transmission delay with the maximum distance is the increase in length of the data transmission route. Therefore, the energy consumed during transmission is as shown in Fig. 11.

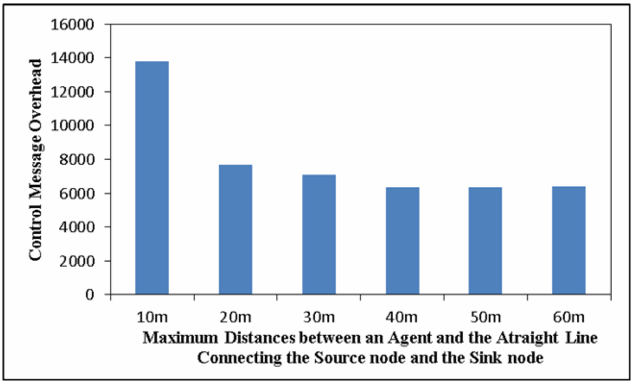

Fig. 9. Control message overhead as a function of the maximum distance between an agent and the straight line connecting the source node and the sink node

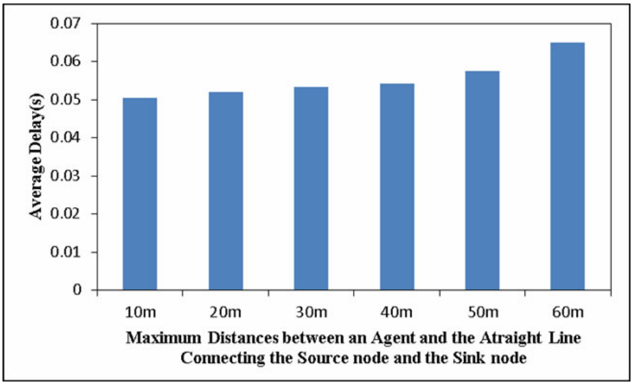

Fig. 10. Comparison of the average transmission delays for various maximum distances between an agent and the straight line connecting the source node and the sink node

From this result, it can be conclude that frequent agent updates caused by a small maximum distance between an agent node and the straight line connecting the source node and the sink node result in high cost. In addition, a very large maximum distance increases the route through which data are transmitted as well as the agent update route, thus increasing energy consumption.

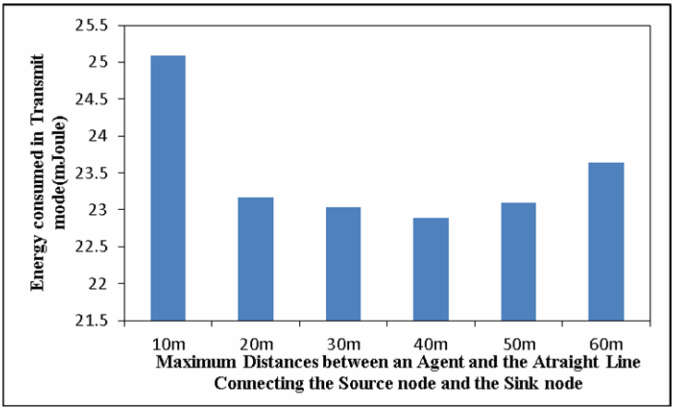

Fig. 11. Comparison of transmission energy consumption for various maximum distances between an agent and the straight line connecting the source node and the sink node

5 Conclusion

We proposed a method in which dynamic agents are used in wireless sensor networks for updating the location information of a sink. In DMALUP, agents are dynamically assigned between the source node and the sink node for data packet transmission, and the last agent is used to update the location information of the sink node. In addition, an agent can undergo dynamic change to create the proximate shortest path, even though it would not be the shortest complete path. In DMALUP, there is a short path for updating the location information of the sink node, and hence, the number of overheads is decreased. Further, the agent is dynamically changed and the shortest proximate route is created, and hence, the delay in data packet transmission is decreased.

The performance of DMALUP was compared with that of elastic routing using the Qualnet 5.0 simulator. In DMALUP, the maximum distance between the agent and the straight line connecting the source node and the sink node was evaluated as well. DMALUP has reduced the control message overhead than elastic routing up to approximately 48%, and has reduced energy consumption up to 32% when velocity is 10*m/s*. Therefore, DMALUP was concluded to be better than elastic routing in terms of control overhead and energy consumption. In DMALUP, when the maximum distance between the agent and the straight line connecting the source node and the sink node was very small, frequent agent changing was required, and hence, a large number of overheads was observed. On the other hand, when the maximum distance was very large, the DMALUP performance decreased owing to the extension of the data packet transmission route.

Acknowledgments

This research is supported by basic science research program through the National Research Foundation of Korea funded by the ministry of education, science and technology (grant No. 2010-0007311). The corresponding author is Professor K. Park.

References

1. Akyildiz, I.F., et al.: Wireless sensor network: A survey. Comp. Networks J. 38(4), 393–422 (2002)
2. Li, J., Jannotti, J., Couto, D., Karger, D., Morris, R.: A scalable location service for geographic ad hoc routing. In: ACM MOBICOM, pp. 120–130 (2000)
3. Mauve, M., Widmer, J., Hartenstein, H.: A survey on position-based routing in mobile ad hoc networks. In: IEEE Network, pp. 30–39 (2001)
4. Karp, B., Kung, H.T.: GPSR: Greedy perimeter stateless routing for wireless networks. In: Proc. of 6th Annual Int'l Conf. on Mobile Computing and Networking, Boston, MA, pp. 243–254 (2000)
5. Ye, F., Luo, H., Cheng, J., Lu, S., Zhang, L.: A two-tier data dissemination model for large-scale wireless sensor networks. In: Proc. of ACM MOBICOM, pp. 148–159 (2002)
6. Wang, G., Wang, T., Jia, W., Guo, M., Li, J.: Adaptive location updates for mobile sinks in wireless sensor networks. J. Supercomputing 47(2), 127–145 (2009)
7. Yu, F., Park, S., Lee, E., Kim, S.: Elastic routing: A novel geographic routing for mobile sinks in wireless sensor networks. IET Communications 4(6), 716–727 (2010)
8. Scalable Network Technologies website, `http://www.scalable-networks.com`