Potential-Based Downstream Routing for Wireless Sensor Networks

Shinya Toyonaga*, Daichi Kominami*, Masashi Sugano† and Masayuki Murata*
*Graduate School of Information Science and Technology, Osaka University, Osaka, Japan
Email:{s-toyonaga, d-kominami, murata}@ist.osaka-u.ac.jp
†School of Knowledge and Information Systems, College of Sustainable System Sciences,
Osaka Prefecture University, Osaka, Japan
Email:sugano@kis.osakafu-u.ac.jp

***Abstract*—In wireless sensor networks designed for periodic monitoring, various many-to-one upstream (sensor-to-sink) routing protocols have been studied. Potential-based routing (PBR) is one such protocol, and can achieve low overhead, high scalability, and efficient load balancing. However, in PBR, unicast messages such as special instructions from a sink node to a certain node are not taken into consideration. In this paper, we propose a potential-based downstream routing protocol (PBDR), in which each sink node constructs an independent potential field and all sensor nodes and sink nodes have a set of potentials determined on each potential field. We refer to the set of potentials as virtual coordinates, based on which we define virtual distance. When a node with downstream data decides a next hop, it calculates the virtual distances from neighbor nodes to the destination node, and forwards the data to the neighbor node closest to the destination node. Through simulation experiments, we show that, given a suitable node density, PBDR attains a data delivery ratio greater than 99.5%. We also show that the data delivery ratio recovers immediately after the failure of 30% of sensor nodes or the failure of a sink node.**

***Keywords*-*sensor networks; potential-based routing; downstream routing; simulation.*

I. Introduction

In wireless sensor networks designed for periodic monitoring, various many-to-one upstream (sensor-to-sink) routing protocols have been studied. In some applications, specific requirements must be met for downstream (sink-to-sensor) data delivery. For example, a sink node sends a query to a specific sensor node upon receiving abnormal data from it, or a sink node sends a message in order to change the frequency of sensing in a specific domain.

Many potential-based routing protocols have been proposed for upstream (sensor-to-sink) data transmission [1-3]. These potential-based upstream routing protocols (PBUR) aim for low overhead, high scalability, and energy balancing. In PBUR, all nodes have a potential. Each node calculates its potential based on local information, such as their neighbor nodes' potentials or residual energy, and a sensor node whose hop count to a sink is smaller (larger) has a higher (lower) potential. Therefore if a node sends data to its neighbor node with higher potential, the data will ultimately reach a sink node. Since these potential fields are constructed on the basis of purely local information, PBUR is scalable. Moreover, if these potential fields are constructed based on residual energy, load balancing can be realized. However, in these protocols, the delivery of downstream (sink-to-sensor) messages such as a special instruction for a certain node is not considered.

In this paper, we propose a potential-based downstream routing protocol (PBDR) for multi-sink wireless sensor networks. In existing PBUR protocols, there is a possibility that some sensor nodes have the same potential since the height of the potential field depends on only the hop count to the sink. Thus, when the sink node transmits data intended for a certain sensor node along the potential field gradient, the data will not always arrive at the destination. In PBDR, multiple sink nodes individually construct potential fields, and all nodes have a set of potentials. The set of potentials are treated as virtual coordinates that identify a destination node. Then, we define a virtual distance between virtual coordinates. A node with data to be sent calculates the virtual distances between the intended destination node and its neighbor nodes, and then forwards the data to the neighbor node closest to the destination node in terms of virtual distance.

We evaluate the data delivery ratio of PBDR at various node densities and packet error ratios, and we use computer simulations to show the protocol's robustness against the failure of multiple sensor nodes or the failure of a sink node.

The rest of this paper is organized as follows. We start by giving an overview of related work in Section II. In Section III, we show the existing potential based routing protocol. We present the proposed PBDR protocol in Section IV. Then in Section V, we evaluate the performance of PBDR through simulation experiments. Finally, Section VI gives our conclusions.

II. Related Work

For wireless sensor networks, various any-to-any routing protocols have been studied. In the flooding method and in the gossiping method, messages are relayed on the basis of broadcasts [4, 5]. These methods suffer from a high number of redundant transmissions, particularly when a few nodes in a specific domain are the destinations.

Many studies have been conducted on proactive and reactive routing protocols [6, 7]. In reactive protocols, each node constructs routes in only the case that communication is required. Then, power consumption can be cut when communication is not needed. The delay time, however, is longer for reactive protocols because of their route discovery procedures. This means that reactive protocols are not appropriate for real-time applications. In proactive protocols, end-to-end delay is small. However, there is overhead because all the nodes collect information about links.

Geographic routing protocols allow for communication between two arbitrary nodes [8]. Some equipment for acquiring the precise geographic position is required for these protocols, and all the nodes must know the position of their destinations. The virtual coordinate assignment protocol (VCap) is able to route data using virtual position without GPS devices [9]. In VCap, all nodes have three shortest hop counts from three anchor nodes and use them as virtual coordinates. Note that since the hop count is an integer, some nodes may have the same virtual coordinate in VCap.

In this paper, we propose a downstream routing based on PBUR. We give an outline of PBUR in the following section.

III. Existing Potential-Based Upstream Routing Protocols

PBUR protocols are categorized as proactive routing protocols. In PBUR, all the nodes have a scalar potential that constructs a potential field. Each node updates its potential based on local information, such as potentials, its residual energy and that of its neighbors, or hop counts to a sink node. A sensor node whose hop count to a sink is smaller (larger) has a higher (lower) potential. Each node with data to be sent forwards the data to a node whose potential is higher than its own, and then the data ultimately reach the sink node. Moreover, load balancing and extending the lifetime of wireless sensor networks by using the residual energy of neighbor nodes or the amount of traffic has been studied [1, 3]. Controlled potential-based routing (CPBR [10]) constructs a potential field for multi-sink wireless sensor networks by using a discrete form of the diffusion equation (1). $\phi(n,t)$ describes the potential of node n at time t. $Z(n)$ is a set of neighbor nodes of node n and $|Z(n)|$ is the cardinality of the set $Z(n)$. A parameter ϵ changes the magnitude of influences by potentials of the neighbor nodes. It is noteworthy that potentials may oscillate when ϵ is larger than one. In CPBR, ϵ is set to the value between 0 and 1 in order to keep the potential from oscillating.

$$\phi(n,t+1) = \phi(n,t) + \frac{\epsilon}{|Z(n)|} \sum_{k \in Z(n)} \{\phi(k,t) - \phi(n,t)\}. \tag{1}$$

In existing PBUR protocols, there is the possibility that some sensor nodes have the same potential. Therefore, when the sink node transmits data to a certain sensor node along the gradient of the potential field constructed through existing PBUR protocols, the data will not always arrive at the destination. This problem is treated as a contour problem as shown in Figure 1. The contour problem is the problem that no node can determine the next hop because no node knows the geographic direction to the destination node by potentials.

In this study, we focus on the advantages of PBUR for wireless sensor networks and implement downstream routing by extending PBUR. As described in subsequent sections, we use the method in CPBR for constructing the potential field, but our method is also applicable to existing PBUR protocols.

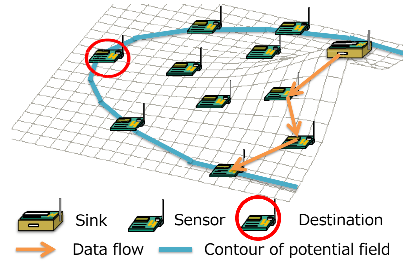


Figure 1. Contour problem for downstream routing using an existing potential construction method

IV. Potential-Based Downstream Routing

PBDR must accomplish the following three tasks in order to handle the contour problem.

1) Assign potentials to all sensor nodes for identifying them
2) Inform the sink nodes of the potentials
3) Route data to a destination node by using its potential as an identifier

In a following PBDR algorithm, we suppose that all sinks can communicate with each other via the local area networks or the Internet.

A. Overview of PBDR

For realizing PBDR, it is first necessary to assign potentials to all sensor nodes in order to identify them. We denote such a potential as P_{id}, and we give an overview of PBDR with P_{id} below.

1) Each sensor node calculates its own P_{id}.
2) When a sensor node generates an upstream data packet, it includes its P_{id} in the packet header, and a sink node records the P_{id} when it receives the upstream data.
3) We define a function $Dist_p(n_1, n_2)$ which is a virtual distance between nodes n_1 and n_2 and is calculated from their P_{id}s.
4) A sensor node with downstream data to be transmitted forwards data to the neighbor node whose distance to the destination node is smallest, as shown by the value of function $Dist_p(n_1, n_2)$. In this way, the data ultimately reaches the destination node.

B. Node Identification

In protocols based on existing methods for constructing a potential field, downstream data will not always arrive at the destination node because of the contour problem. Thus, we assign a virtual coordinate to all sensor nodes in order to identify them. This method is based on the idea of the trilateration. N sink nodes individually construct potential fields, and all nodes have a set of potentials as a virtual coordinate. Here, as in reference [10], the diffusion equation is used by sink node i to construct the potential field F_i $(i = 1, \cdots, N)$. Now, we can define that P_{id} is a set of N potentials. If there are three sink nodes and three potential fields, PBDR can be realized. However, in Section V, we use four sink nodes and four potential fields in order to acquire the redundancy when a sink node fails.

Equation (2) is used to construct the potential field F_i having potential $\phi(n,t,i)$ at node n and time t. ϵ is a constant which plays the same role as the ϵ in the equation (1).

$$\phi(n,t+1,i) = (1-\epsilon)\cdot\phi(n,t,i) + \frac{\epsilon}{|Z(n)|}\sum_{k\in Z(n)}\phi(k,t,i). \quad (2)$$

Generally, in the diffusion equation, when all boundary conditions have the same value, all values in the field converge on the value of the boundary conditions, and the field eventually becomes flat. Consequently, potential routing does not work because there is no gradient in the field without a boundary condition. So, we use equation (3) as a boundary condition so that the potentials of the entire network do not converge on the potential of a sink node. S is a set of sink nodes. Note that sink node i constructs the potential field F_i.

$$\forall s \in S, \phi(s,t,i) = \begin{cases} \phi_{max} & \text{if } i = s \\ \phi_{min} & \text{otherwise.} \end{cases} \quad (3)$$

C. Downstream Routing

To send a data to a specific sensor node, sink nodes must know P_{id} of a destination node. In this paper, we assume that every sensor node uses the potential field whose potential is the highest in its P_{id} in order to periodically send a monitored data packet to the nearest sink node. Each sensor node inserts its P_{id} into the header of the packet and sink nodes can collect P_{id} for every sensor node. In this manner, downstream routing is realized by simultaneously performing upstream routing.

We assume that all the sink nodes are connected to each other with a high speed link. Also, a downstream data packet can be routed to the sink node closest to a destination node and the sink node can start delivery of downstream data.

We define potential distance as a virtual distance calculated from P_{id}. To select a next hop, node n calculates the potential distance $Dist_p$ between its neighbor $k(\in Z(n))$ and destination node d:

$$Dist_p(k,d) = \sqrt{\sum_{i=1}^{N}\left(F_i\left(k\right) - F_i\left(d\right)\right)^2}. \quad (4)$$

$F_i(k)$ is the potential of node k on the potential field F_i, and $F_i(d)$ is the potential of destination node d. We use potential distance as a routing metric.

A sink node includes P_{id} of destination node d in the header of a downstream data packet, and relay nodes forward the data to node y that fulfills the following condition:

$$y = \arg\min_{k\in Z(n)} Dist_p(k,d). \quad (5)$$

When a data reaches a neighbor node of destination node, the data is forwarded to the destination node using a node ID.

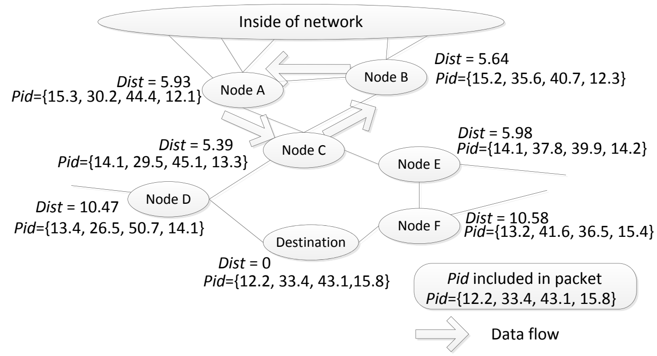


Figure 2. Local-minimum problem

D. Local-Minimum Problem

When a destination node is in a domain that has low node density, for example, at the boundary of a monitoring area, the local-minimum problem may occur. The local-minimum problem arises when a node cannot forward data anywhere, because low node density can lead to void areas and there is no neighbor node closer to the destination node in terms of potential distance.

In the example shown in Figure 2, node C must forward data to node D so that the destination node receives the data. However, $Dist_p(\text{node } D, \text{destination})$ is larger than $Dist_p(\text{node } B, \text{destination})$ and node C does not forward data to node D. We use a local detour rule, in which node v forwards data to node w having the smallest $Dist_p(w, \text{destination})$, even if $Dist_p(v, \text{destination})$ is smaller than $Dist_p(w, \text{destination})$, and node v does not forward data to node u after node v receives the data from node u. According to this rule, node C forwards the data to node B. As a result, a data packet will follow a loop through node A, node B, and node C.

The local-minimum problem occurs when a destination node is near the boundary of the monitoring area, where the node density is low and there is a void area. So we assume that a destination node is near the boundary of the monitoring area when a loop is detected, and we resolve the local-minimum problem by using an alternative routing metric.

Because the height of potential can be treated as a virtual distance from a sink node, a loop is hard to occur when a single potential field is used for downstream routing. So, we use a single potential field when a loop is detected. The node near the monitoring area boundary is located in the area farthest from a certain sink node and the potential of the destination node on the potential field built by the sink node is nearly equivalent to $\phi_{\min}$. Thus, a possibility that the data packet gets close to the boundary of the monitoring area is high when a node forwards the packet to the node farthest from the sink node. In order to send data to the boundary of the monitoring area, we use only one potential field whose potential is the smallest in P_{id} of the destination node. From the above, we define a potential gap $Gap(k,d)$ (6), and use it as an alternative routing metric when a routing loop is detected. Node d is a destination node, and node k is a neighbor of the node that detects a routing loop. The node that detects the routing loop forwards the data to the node whose potential gap with respect to the destination node is the smallest.

$$Gap(k, d) = |F_i(k) - F_i(d)|, i = \arg\min_{1\leq j\leq N} F_j(d). \quad (6)$$

For example, in Figure 2, potential gap of $\mathrm{Node}\ A$ is 3.1, one of $\mathrm{Node}\ C$ is 1.9 and one of $\mathrm{Node}\ D$ is 1.2. Then, $\mathrm{Node}\ C$ forwards the data to $\mathrm{Node}\ D$ and the data reaches the destination node.

A sequence number and a loop flag are included in the data packet header and are used to detect routing loops. When a node receives a downstream data packet, the node records the sequence number of the data. When a node receives data with the same sequence number, the node judges that a loop has occurred and sets the loop flag to 1. Each node records $n_{history}$ sequence numbers of received packets from the newest one. The routing protocol is shown below for when a loop is detected.

1) Node n checks whether it has recorded a sequence number of the data. If node n detects a loop, it executes process 2. Otherwise, it executes process 3.
2) Node n sets a loop flag to 1 indicating that the data packet is in a loop and executes process 4.
3) Node n sets a loop flag to 0 indicating that the data packet is not in a loop and executes process 4.
4) When a node receives a data packet destined for another node, the receiving node checks this flag in the data. If the flag is set to 1, Gap is used; if the flag is set to 0, $Dist_p$ is used. However, if the flag is set to 1 and there is no node that has a potential gap of less than a potential gap the receiveing node has with respect to the destination node, the receiving node clears the flag.

E. MAC Layer Protocol

In this paper, we use intermittent receiver-driven transmission (IRDT) for the MAC layer protocol [11]. In IRDT, all nodes sleep and wake up asynchronously with the duty cycle $T_{dutycycle}$. Whenever a node wakes up, it sends an ID message that informs neighbor nodes that the node is ready to receive data.

When node n_1 forwards data to node n_2 in IRDT, the procedure shown in Figure 3 is used. Node n_1 with data to be sent wakes up and waits for an ID message from node n_2. Upon receiving an ID message from node n_2, node n_1 sends an SREQ message informing node n_2 that it has a data packet for node n_2. When node n_2 receives the SREQ message, it stays awake and sends to node n_1 a RACK message, which is an acknowledgement for communication request. After that, node n_1 sends a data message to node n_2. Finally, node n_2 sends to node n_1 a DACK message, which is an acknowledgement for data. If node n_2 is not a destination node, node n_2 becomes a sender and waits for an ID message from a neighbor node.

Node n_1 drops the data when forwarding the data does not succeed within $T_{timeout}$ after node n_1 woke up. Also, when the number of forwards exceeds time to live (TTL), node n_1 drops the data.

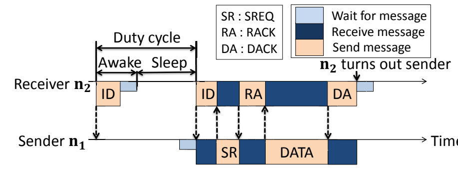


Figure 3. Procedure for forwarding data in IRDT

Table I
SIMULATION CONFIGURATION

Parameter	Value
Radio range	100 m
Time to live (TTL)	15
Data packet size	128 byte
Bandwidth	100 kbps
$\phi_{\max}$	90
$\phi_{\min}$	0
ϵ	0.8
$T_{dutycycle}$	1 s
T_{update}	50 s
$T_{timeout}$	5 s
$n_{history}$	3

V. SIMULATION EXPERIMENTS

In this section, we present the results of our simulation experiments. PBDR is implemented in the OMNeT++ [12] network simulator. We evaluate data delivery ratio of PBDR at various node densities and packet error ratios, and show its robustness against sensor-node failure and sink-node failure.

The sensor nodes are randomly distributed in a 600 m $\times$ 600 m square. In this network, the number of deployed sensor nodes is from 50 to 250 and 4 sink nodes are situated at the four corners of observation area. In the data generation model, the rate of data generation is $\frac{1}{100}$ per node for upstream communication and $\frac{1}{300}$ per node for downstream communication in Poisson process. The reason that the rate of data generation is higher for upstream communication than for downstream communication is that downstream communication is demanded less frequently. Under these conditions, we evaluate how the data delivery ratio is affected by node density and packet error rate (Section V-A), by sensor nodes failure (Section V-B), and by sink node failure (Section V-C). The details of the simulation configuration are summarized in Table I.

A. Node Density

Simulation results are shown in Figure 4 for changing the packet error rate from 0 to 0.4. The horizontal axis shows the number of nodes, and the vertical axis shows the data delivery ratio. The number of trials is 50, and the confidence interval is 95%.

The data delivery ratio is low when the node density is low because there are few links in the entire network and the local-minimum problem easily occurs. The data delivery ratio increases when the node density is high because the number of links in the entire network increases. When the node density is excessively high, however, packet collisions occur frequently, thus decreasing the data delivery ratio. The data delivery ratio is highest when the number of nodes is 150. In that case, the data delivery ratio is 99.5% and the average number of neighbor nodes is 16.7.

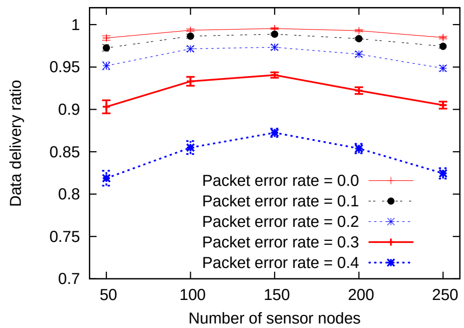


Figure 4. Data delivery ratio vs node density

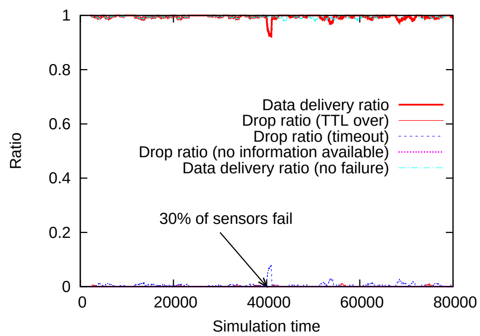


Figure 5. Data delivery ratio and drop ratio (sensor nodes failure)

B. Failure of Sensor Nodes

In the case that there are 150 nodes and 45 sensor nodes fail, we evaluate the data delivery ratio from $t - 1000$ s to t at each time t. The simulation time is 80,000 s and 45 sensor nodes fail after 40,000 s have elapsed. The number of trials is 10.

Figure 5 shows the data delivery ratio and drop ratio from $t - 1000$ s to t at each time t. The drop ratio (TTL over) is the packet drop ratio when the number of forwards of the data is over TTL. The drop ratio (timeout) is the packet drop ratio when a node with a data packet cannot forward the data within $T_{timeout}$ after the node wakes up. The drop ratio (no information available) is the packet drop ratio when no sink node has P_{id} of a destination node. The results show that downstream routing works well even if sensor nodes fail.

The data delivery ratio decreases steeply when sensors fail at 40,000 s, but quickly recovers to the level observed before node failure. The drop ratio (TTL over) and no information available do not change considerably when sensor nodes fail, but the drop ratio (timeout) increases steeply. After node failure, data packets are dropped more frequently by time out. This is because the number of links in the entire network decreases and the load on the entire network increases after sensor nodes fail.

C. Failure of Sink Node

In the case that there are 150 nodes and one of the four sink nodes fail, we evaluate the data delivery ratio at immediately prior to 1,000 s. The simulation model is the same as one for sensor nodes failure. When sink node s fails, all the potentials of F_s converge on $\phi_{\min}$ because of boundary condition 3. A sensor node with upstream data to be sent decides the next hop according to the potential field whose value is highest among the potentials. In this manner, the other three sink nodes collect P_{id} for each sensor node and PBDR regains its effectiveness after sink node failure.

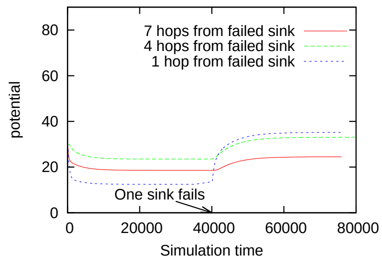
Figure 6 shows changes in potential until the potential fields converge. Here, the changes in potential for three nodes are shown. The first is the farthest from the failed sink node, with a hop count to the failed sink node of 7. The second is deployed near the center of the network, with a hop count to the failed sink node of 4. The third is a 1-hop neighbor of the failed sink node. In Figure 6(b), the changes of the potential field that were constructed by the failed sink node are shown and the potentials converge on $\phi_{\min}(= 0)$ in about 30,000 s. In Figures 6(a), 6(c), and 6(d), the changes in the potential fields that the other three sink nodes construct respectively are shown and the potentials converge in about 20,000 s.

The data delivery ratio and drop ratio at just before 1,000 s in each cases is shown in Figure 7. These results show that downstream routing works well, even if one of four sink nodes fails. When a sink node fails at 40,000 s, the data delivery ratio sharply decreases temporarily and increases again. This is because the other three sink nodes lack the P_{id} of sensor nodes which failed sink node held, and the drop ratio (no information available) become temporarily high. After a sink node fails and the other three sink nodes collect potential information about all sensor nodes, the drop ratio (no information available) decrease. However, data are dropped by time out more frequently. This is because, when a sink node fails, it becomes necessary for the upstream flow to be processed by three sink nodes instead of four and the load on the entire network increases accordingly. We note that the data delivery ratio increases more quickly than the time of potential convergence which is about 20,000s.

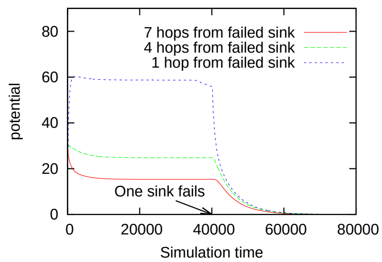
VI. Conclusion and Future Work

In PBDR, multiple sink nodes construct independent potential fields and all nodes have a set of potentials used as a virtual coordinate. We defined virtual distance based on virtual coordinates and use it as a routing metric. Through OMNeT++ simulation, we evaluated the data delivery ratio for various node densities and packet error rates, as well as the robustness against failure of multiple sensor nodes or of a sink node. PBDR achieves a data delivery ratio greater than 99.5% when the network has a suitable node density. Even if multiple sensor nodes fail or a sink node fails, the data delivery ratio recovers immediately after sensor-node failure or sink-node failure.

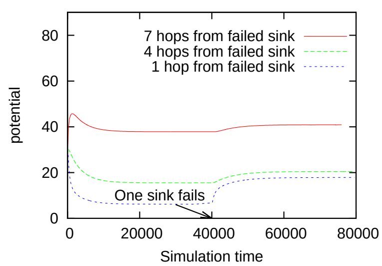
In PBDR, when the number of potential field increases, reliability of downstream routing can be raised, but overhead also increases. Hence, we plan to investigate this tradeoff in future work. In this paper, we do not consider power consumption. So, we will evaluate the distribution of power consumption using the method for constructing the potential field with load balancing.



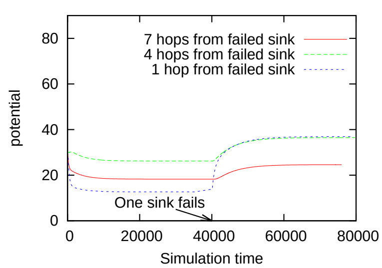
(a) potential F_0



(b) potential F_1 (constructed by failed sink node)



(c) potential F_2



(d) potential F_3

Figure 6. Potential convergence after sink-node failure

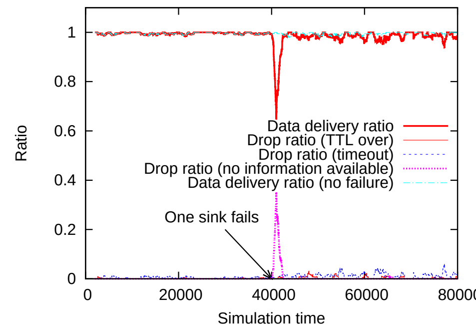


Figure 7. Data delivery ratio and drop ratio (sink node failure)

Acknowledgment

This research was supported in part by "Grant-in-Aid for Scientific Research (C) 23500097 and for JPSP Fellows (24738)" of the Japan Society for the Promotion of Science (JSPS) in Japan.

References

[1] C. Wu, R. Yuan, and H. Zhou, "A novel load balanced and lifetime maximization routing protocol in wireless sensor networks," in *Proceeding of IEEE Vehicular Technology Conference*, May 2008, pp. 113–117.

[2] A. Basu, A. Lin, and S. Ramanathan, "Routing using potentials: a dynamic traffic-aware routing algorithm," in *Proceedings of ACM SIGCOMM 2003*, Aug. 2003, pp. 37–48.

[3] H. Liu, Z.-L. Zhang, J. Srivastava, V. Firoiu, and B. DeCleene, "PWave: flexible potential-based routing framework for wireless sensor networks," in *Proceeding of IFIP/TC6 Networking Conference*, May 2007, pp. 14–18.

[4] S. Guo, Y. Gu, B. Jiang, and T. He, "Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links," in *Proceedings of ACM MobiCom 2009*. New York, NY, USA: ACM, Sep. 2009, pp. 133–144.

[5] S. Fauji and K. Kalpakis, "A gossip-based energy efficient protocol for robust in-network aggregation in wireless sensor networks," in *Proceedings of IEEE Pervasive Computing and Communications Workshops*, Mar. 2011, pp. 166 –171.

[6] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum, and L. Viennot, "Optimized link state routing protocol," *Internet Draft draft-ietf-manet-olsr-txt Work in progress*, pp. 1–15, Apr. 2003.

[7] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings of IEEE Mobile Computing Systems and Applications*, Feb. 1999, pp. 90–100.

[8] L. Shu, Y. Zhang, L. Yang, Y. Wang, M. Hauswirth, and N. Xiong, "TPGF: geographic routing in wireless multimedia sensor networks," *Telecommunication Systems*, vol. 44, pp. 79–95, Oct. 2010.

[9] A. Caruso, S. Chessa, S. De, and A. Urpi, "GPS free coordinate assignment and routing in wireless sensor networks," in *Proceedings of IEEE INFOCOM 2005*, Mar. 2005, pp. 150–160.

[10] D. Kominami, M. Sugano, M. Murata, and T. Hatauchi, "Controlled potential-based routing for large-scale wireless sensor networks," in *Proceedings of Modeling, analysis and simulation of wireless and mobile systems*, Jan. 2011, pp. 187–196.

[11] C. Damdinsuren, D. Kominami, M. Sugano, M. Murata, and T. Hatauchi, "Lifetime extension based on residual energy for receiver-driven multi-hop wireless network," in *Cluster Computing*, May 2012. [Online]. Available: http://dx.doi.org/10.1007/s10586-012-0212-0 [retrieved: October, 2012]

[12] A. Varga, "OMNeT++," in *Modeling and Tools for Network Simulation*. Springer Berlin Heidelberg, Oct. 2010, pp. 35–59.