An Energy-Efficient *k*-Disjoint-Path Routing Algorithm for Reliable Wireless Sensor Networks

Jang Woon Baek[1], Young Jin Nam[2], Dae-Wha Seo[1]

[1] School of Electrical Eng. & Computer Science, Kyungpook National University
[2, *] School of Computer & Information Technology, Daegu University
[1] {kutc, dwseo}@ee.knu.ac.kr, [2]yjnam@daegu.ac.kr

Abstract. Wireless sensor networks are subject to sensor node and link failures due to various reasons. This paper proposes an energy-efficient, *k*-disjoint-path routing algorithm that adaptively varies the number of disjoint paths (*k*) according to changing data patterns and a target-delivery ratio of critical events. The proposed algorithm sends packets through a single path (*k*=1) under no occurrence of critical events, whereas it sends through *k* disjoint paths (*k*>1) under the occurrence of critical events, where *k* is computed from a well-defined fault model and the target delivery ratio. Note that the proposed algorithm detects the occurrence of critical events by monitoring changing data patterns. Our simulations reveal that the proposed algorithm not only guarantees the target delivery ratio as much as a multi-path routing algorithm, but also makes energy consumption and average delay as low as a single-path routing algorithm.

Keywords: Wireless sensor networks, data variation, disjoint-path routing

1   Introduction

According to advances in MEMS, wireless communication, and digital electronics technology, wireless sensor networks have been widely deployed to monitor and control physical environments [1]. Wireless sensor networks typically consist of a large number of sensor nodes which can observe physical phenomena, process sensed information, and communicate with other nodes. Since sensor nodes are equipped with limited battery power, low-power consumption is very crucial in wireless sensor networks [2]. It is believed that power consumption is dominated by the costs of transmitting and receiving messages [3]. It is known that in-network aggregation can save a significant amount of energy by reducing the number of transmitted messages over wireless sensor networks [4]. Sensor nodes with in-network aggregation can combine data from their child nodes and their locally-collected data before sending a message to their parent nodes. Typically, in-network aggregation employs a single-path routing algorithm with a tree topology for energy saving [4, 10]. However, if any node on a single routing path fails, the data packet cannot be delivered to the base

---

station. Actually, individual sensor nodes are highly vulnerable to failures caused by battery drain, outside damages, or security attacks [5].

Multi-path routing algorithms have been proposed for the reliable event delivery in wireless sensor networks [6-8]. Basically, this breed of routing algorithms constantly employs m disjoint paths, where m is generally determined in an ad-hoc manner. With a larger m, the multi-path routing algorithms require more nodes to participate in the event transmission. In result, more energy is consumed, and the overall traffic in the wireless sensor network is increased, thus resulting is a higher possibility of congestion and worse load balancing [9]. Another breed of routing algorithms for reliable event delivery is the path-repair routing algorithm that forwards data along a single path and repairs paths in the presence of failures in order to achieve a higher delivery ratio with low-energy consumption [9, 10]. Path-repair routing algorithms, however, usually generate additional latency in the search for alternative paths. If some packets arrive after a timeout especially during in-network aggregation, the aggregation process excludes these packets. To make matters worse, if packets encompass any critical events, packet loss poses serious problems for sensor applications.

This paper proposes an energy-efficient k-disjoint-path routing algorithm for in-network aggregation over wireless sensor networks. The key of the proposed algorithm is to adapt the number of disjoint paths (k) according to the changing data patterns and the target delivery ratio of critical events, such as fires and poisonous gas leaks, etc. The proposed algorithm configures k=1 if the variance of the received data (data variation) from children is lower than a pre-defined threshold, whereas it sets k>1 if data variation is higher than the pre-defined threshold, where k is determined by the target delivery ratio. Note that the proposed algorithm detects the occurrence of critical events by monitoring data variation. Since sensor nodes usually collect non-critical events that have less important information and little effect on aggregation results, the proposed algorithm spends a large portion of its lifetime operating like the single-path routing algorithm. As a result, the proposed algorithm consumes much less power than multi-path routing algorithms. After detecting the occurrence of critical events, the proposed algorithm begins to work like a k-disjoint-path routing algorithm in order to meet target delivery ratios.

The remainder of this paper is organized as follows. Section 2 provides background information on multi-path routing and path-repair algorithms. Section 3 offers a detailed description of the proposed algorithm, and Section 4 compares the performance of the proposed algorithm with that of exiting routing algorithms. Finally, concluding remarks are presented in Section 5.

2   Background

Data collection from wireless sensor networks typically uses a single-path routing algorithm with tree topology for energy saving [4, 11]. Wireless sensor networks have a relatively short radio range and may be deployed into a large geographical coverage area, i.e., the route between a source node and a base station is likely to consist of a large number of hops. As a result, the success probability of the single-path routing

becomes very low. There are two different methods to maintain routing paths in the presence of node failures: multi-path routing algorithms and path-repair algorithm.

Multi-path routing algorithms have shown higher resilience to node failure both theoretically and experimentally comparing with the single-path routing [6-8]. The construction of disjoint multi-path is described in [7]. In this approach, multiple copies of data are sent along different paths, allowing for resilience to failures of a certain number of paths. For instance, the same data packet along m disjoint paths can increase the delivery ratio in approximate proportion to m, as compared with the single-path routing algorithm. A smaller value of m can save energy, but it is less likely to meet a target delivery ratio. On the other hand, a larger value of m is more likely to guarantee a target delivery ratio, whereas it not only causes higher energy consumptions, but it also creates more traffic for the packet delivery. Note that more traffic implies higher collisions in wireless channels and longer back-off delays for transmission.

Path-repair algorithms are proposed to overcome the problems of multi-path routing algorithms [9, 10]. In the face of path failure, a notification is sent to the source node, which is responsible for finding an alternative path and resending the packet. However, this kind of a source-initiated path-repairing approach is inefficient, especially when a failure occurs in many hops away from the source node. To decrease long-path recovery time and unnecessary energy consumption of source-initiated path-repairing approaches, local-node-based path-repair algorithms have been proposed [10]. These algorithms detect packet loss with implicit acknowledgements [9]. In the presence of path failure, a sensor node searches for an alternative node among a list of neighbor nodes, and it immediately sends the packet stored in its local cache to the alternative node. These algorithms, however, still make additional latency while discovering alternative paths.

3   The Proposed Algorithm

To begin, we assume that data aggregation is performed periodically in a wireless sensor network. The key of the proposed algorithm is to vary the number of disjoint paths (k) according to the changing data patterns and the target delivery ratio of critical events. Note that the routing algorithm operates at each sensor node in a distributed manner.

The proposed algorithm exploits the following characteristics of sensor readings in order to detect critical events. First, as time passes, there is little change in sensor readings from wireless sensor networks. Second, there is little change in sensor readings from physically-adjacent sensor nodes in a wireless sensor network. Third, in the presence of critical events, such as earthquakes or forest fires, there is significant change in sensor readings from a wireless sensor network. By using the first two characteristics, the proposed algorithm maintains single-path routing (k=1) when a variance of the received data from children is lower than a pre-defined threshold because there are not any critical events. As a result, it can consume energy dramatically compared with the multi-path routing algorithms. The proposed algorithm, however, employs the k-disjoint path (k>1) when the variance becomes

higher than a pre-defined threshold in order to meet a target delivery ratio of critical events.

The occurrence of a critical event can be detected by monitoring data variation. For example, a sharp change in the light intensity, an unusual sound matching a certain signature and a very high temperature in the same region within a specified time interval implies an explosion event [12]. In building risk monitoring, if the variation of data collected by acceleration and strain sensors of sensor node is large, we can infer that an abnormal symptom of building is generated [13]. In the earthquake detection, a sensor node can detect the event using an earthquake detection algorithm triggered on significant seismic or acoustic signal [14]. From theses examples, we can see that data variation is important to detect the occurrence of critical events. The information on detected critical events has to be reliably transferred to the base station.

The threshold of data variation is set by using maximum data variation in a normal condition without occurrence of critical events. The normal data variation can be empirically obtained by running iterative experiments. Data variation depends on the location of sensor node from the point of event occurrence. We can change the threshold of data variation according to the characteristics of sensor applications.

The proposed algorithm computes the minimum number of disjoint paths ($k>1$), say k', between a sensor node and its final destination (a base station) in the presence of critical events to meet a given target delivery ratio. Presently, it uses a simple fault model, where each sensor node can be faulty independently with the probability of f in a wireless sensor network. Fig. 1 shows a packet-transmitting model through k disjoint multiple paths at a sensor node located N hops away from the base station. The probability that no duplicated packet is delivered to the base station, $P_{fail}(N)$, is calculated as $P_{fail}(N) = (1 - (1 - f)^N)^k$. Therefore, the probability that at least one single copy of the packet is delivered to the base station can be computed as $P_{succ}(N) = 1 - P_{fail}(N)$. Given a target delivery ratio, $P_{succ}(N) = \alpha$, the minimum number of disjoint paths denoted by k' can be computed as

$$k' = log(1 - \alpha) / log(1 - (1 - f)^N). \tag{1}$$
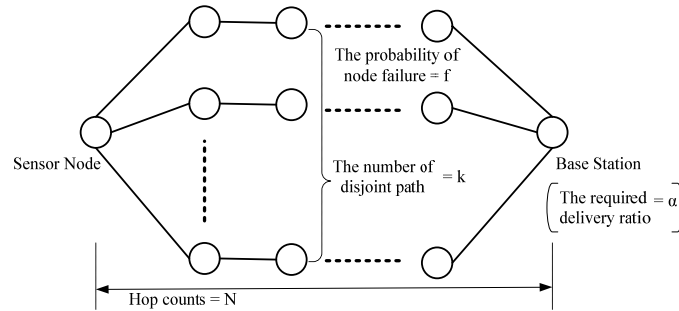
Fig. 1. Packet-transmitting model through k disjoint paths

```
procedure PATH_INITIALIZATION()
    // N = {n| node(n)},  all sensor nodes
    // Pn = {pi}i=1, …, M,  parent list of node(n)

    foreach sensor node(n) ∈ N do
        receive queries from its parents;
        initialize a parent list Pn ;
        compute the number of disjoint path k' to meet
                         the target delivery ratio(α);
        construct k' disjoint paths to the base station;
        pick k' parents and its primary parent;
    end

end PATH_INITIALIZATION

procedure DATA_TRANSMISSION()
    n ← node_id;
    packetn ⇐ sensing critical or non-critical events;

    repeat
        packetchild(n) ← receive(nodechild(n));
        packetn ← data_aggregation(packetn, packetchild(n));
        update data variation(dvn) at noden;
        check threshold conditions and update Timeoutn
    until Timeoutn has expired

    if (critical event occurred) then
        send packetn to each of the k' parents;
    else
        send packetn to the primary parent;
    endif

    sleep until the next period;
end DATA_TRANSMISSION
```

Fig. 2. The operational behavior of the proposed algorithm

The operational behavior of the proposed algorithm is described in Fig. 2. We assume that there exits a set of sensor nodes denoted by N, where each sensor node is denoted by a $node_n$. We denote the parent nodes and the child nodes of a $node_n$ by $node_{par(n)}$ and $node_{child(n)}$, respectively. The first step of the proposed algorithm (**PATH_INITIALIZATION**) is to construct a minimum number of disjoint paths between each sensor node to the base station. This is done in order to meet a given target delivery ratio of critical events. A sensor node receives multiple queries from its parent nodes ($node_{par(n)}$) and initializes its parent list (P_n) and data aggregation parameters including a target delivery ratio (α) and a data variation threshold (DV). Next, the proposed algorithm calculates the minimum number of disjoint paths (k') by using Eq.(1) and constructs k' disjoint paths from the $node_n$ to the base station as follows. The $node_n$ sends duplicate requests including information of *source ID*, *sender ID, hop counts,* and *request ID* to k' parents that are randomly selected from the parent list. Note that the request ID is unique during toward the base station. When a node at a lower hop count receives the request, it stores the path information unless it has received the same request ID. Otherwise, it returns the NACK message to the request sender. Then, the sender delivers the request message to another parent. By repeating this process at each node in a wireless sensor network, the base station can receive the request messages through k' disjoint paths. The base station returns acknowledgements to the $node_n$ through each of the reverse paths for the confirmation of the path setup. Finally, path initialization chooses k' parents for k'-disjoint-path routing and a primary parent is selected from the k' parents.

The next step in the proposed algorithm (`DATA_TRANSMISSION`) is to transmit sensed data to the base station. The $node_n$ waits for packets ($packet_{child(n)}$) from the child nodes, and fulfills partial aggregation; the $node_n$ checks the threshold condition of data variation during a timeout (T_n), which is configured by the adaptive timeout scheduling scheme [10]. If data variation (dv_n) at $node_n$ is within DV, which means that critical events have not occurred, the proposed algorithm sends a packet to its primary parent (single-path routing). If the data variation is larger than DV, which means that critical events have occurred, the proposed algorithm sends a packet to each of the k' parents in the parent list (k'-disjoint-path routing). When the transmission is completed, the $node_n$ sleeps until the next period.

4 Performance Evaluations

This section evaluates the performance of the proposed algorithm via simulations. We implemented the proposed k-disjoint-path algorithm in the ns-2 network simulator [16]. We compared the proposed algorithm with a single-path routing algorithm [4], a multi-path routing algorithm [7], and a path-repair routing algorithm [9]. The performance metrics encompass the average dissipated energy, the event delivery ratio, and the average delay.

Table 1. Simulation parameters

Parameters	Values
Number of nodes	100
Transmission Range	50m
Physical Link Bandwidth	1Mbps
MAC Layer	802.11
Transmit Energy	14.88mW
Receive Energy	12.50mW
Data Collection Period	1 sec.
Critical Event Injection	50 sec.
Data Variation Threshold	5%
Probability of A Node Failure (f)	0.1

4.1 Simulation Environment

Table I shows the parameters used in our experiments. Each simulation begins by deploying 100 sensor nodes randomly in a 1000-by-1000 grid. All the sensor nodes remain static after being deployed. The radio range of each node is set to 50 meters. The bandwidth of each physical link is 1Mbps. We employ an 802.11 MAC-layer

protocol, where the transmission and reception energy consumption is set to 14.88 and 12.50mW [15]. Each simulation runs for 1,000 seconds, and the sensor application collects sensed data every one second through the wireless sensor network. Critical events are randomly injected into sensor nodes in the wireless sensor network. The threshold of data variation to detect the critical events is set to 5 percents. In normal environments, the data variation of temperature is lower than 5% at the reasonable reporting period. We assume that each node can be faulty independently with the probability ($f=0.1$), and the sensor application demands the target delivery ratio of 0.9 ($\alpha=0.9$).

4.2   The Average Dissipated Energy

The average dissipated energy represents the average of the total dissipated energy at each node when the simulation completes. Table 2 summarizes the average dissipated energy of the proposed algorithm and the other routing algorithms with $f=0.1$ and $\alpha=0.9$. The number of the disjoint paths, 5-disjoint path, for the multi-path routing algorithm is obtained from Eq.(1). The proposed algorithm is observed to consume 43% more energy than the single-path routing algorithm. However, it consumes much less energy than the multi-path routing algorithm by 110% and even the path-repair routing algorithms by 47%. Recall that the path-repair routing algorithm generates additional packets to find an alternative routing path.

Table 2.   A comparison of the average dissipated energy with the different routing algorithms: The single-path algorithm, the multi-path (5-disjoint path) algorithm, the path-repair algorithm, and the proposed algorithm, where $f=0.1$ and $\alpha=0.9$

Routing Schemes	Single-path	Multi-path	Path-repair	Proposed
Dissipated Energy (J)	1.013	2.616	1.822	1.239
Improvement (%)	158.0	-	43.5	110.0

## 4.3   The Event Delivery Ratio of Critical Events

The event delivery ratio of critical events represents the ratio of the number of the critical events received at the base station to the total number of critical events injected at sensor nodes. Fig. 3 depicts the results of the event delivery ratio of critical events with the different routing algorithms. While the single-path routing algorithm cannot guarantee the target delivery ratio of 90%, the multi-path and the proposed routing algorithms can meet the target ratio. Interestingly, the path-repair provides only the event delivery ratio of 82% that is slightly lower than the target ratio. Additional experiments show that the multi-path routing algorithm with three disjoint paths cannot meet the target ratio, whereas four disjoint paths can barely guarantee the target ratio.
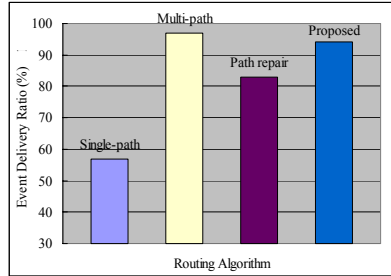
Fig. 3. A comparison of the event delivery ratio of critical events with the different routing algorithms (max. 10-hop distance between the sensor node and the base station): The single-path algorithm, the multi-path (5-disjoint path) algorithm, the path-repair algorithm, and the proposed algorithm, where f=0.1 and α=0.9

4.4 The Average Delay

The average delay represents the average latency required to transmit a critical event from a sensor node to the base station. Fig. 4 shows the average delay measured with each routing algorithm. The proposed algorithm has about 15% longer average delay compared with the single-path routing algorithm making the shortest average delay. However, the average delays of the path-repair routing algorithm and the multi-path routing algorithm are higher than that of the proposed algorithm by 22% and 37%, respectively. Again, the path-repair routing algorithm requires a considerable amount of time to search an alternative path and retransmit the packet. The multi-path routing generates more traffic by sending a packet along multiple-disjoint paths. As a result, it causes higher collisions in wireless channels and longer back-off delays for transmission.
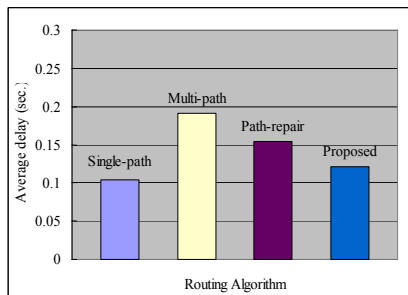
Fig. 4. A comparison of the average delay with the different routing algorithms (max. 10-hop distance between the sensor node and the base station): The single-path algorithm, the multi-path (5-disjoint path) algorithm, the path-repair algorithm, and the proposed algorithm, where f=0.1 and α=0.9

5 Concluding Remarks

This paper proposed an energy-efficient k-disjoint-path routing algorithm that can vary the number of disjoint paths (k) according to changing data patterns and a target delivery ratio of critical events. The proposed algorithm operates as in the single-path routing algorithm ($k=1$) if the data variation is lower than a pre-defined threshold under no occurrence of critical events in order to consume less energy. On the other hand, it works like a multi-path (k-disjoint path) routing algorithm ($k>1$), if data variation is higher than the pre-defined threshold in order to meet the target delivery ratio under the occurrence of critical events. The value of k is obtained from the well-defined fault model and a target delivery ratio. Our simulation showed that the proposed algorithm guaranteed the target delivery ratio as much as the multi-path routing algorithm, whereas the single-path and path-repair routing algorithms could not meet the target ratio. Compared with the single-path routing algorithm, the proposed algorithm should a slight increase in the average dissipated energy and the average delay by 18% and 15%, respectively. The multi-path routing algorithm and the path-repair algorithm, however, consume more energy than the proposed algorithm by 110% and 47%, respectively. In future work, we plan to apply more complex fault models to the current routing algorithm for reliable wireless sensor networks.

References

1. P. Levis, S. Madden, D. Gay, J. Polastre, R. Szewczyk, A. Woo, E. Brewer, and D. Culler: The Emergence of Networking Abstractions and Techniques in TinyOS. In First USENIX/ACM Symposium on Networked Systems Design and Implementation (2004)
2. I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci: Wireless sensor networks: A survey. Computer Networks, Vol. 38 (2002)
3. Y. Yao, and J. Gehrke: Query Processing for Sensor Networks. In Proceedings of CIDR (2003)
4. S. Madden, R. Szewczyk, M. Franklin and D. Cullera: Supporting Aggregate Queries Over Ad-Hoc Wireless Sensor Networks. In Proceedings of WMCSA (2002)
5. J. Deng, R. Han, S. Mishra: A Robust and Light-Weight Routing Mechanism for Wireless Sensor Networks. In Proceedings of DIWANS (2004)
6. F. Ye, G. Zhong, S. Lu and L. Zhang: GRAdient Broadcast: A Robust Data Delivery Protocol for Large Scale Sensor Networks. Wireless Networks, Vol. 11 (2005)
7. D. Ganesan, R. Govindan, S. Shenker, and D. Estrin: Highly-Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks. Mobile Computing and Communication Review, Vol 4, No. 5 (2001)
8. C. Karlof, Y. Li, and J. Polastre: ARRIVE: Algorithm for Robust Routing in Volatile Environments. Technical Report, UCB//CSD-03-1233 (2003)

9. M. Ortolani, L. Gatani, and G. Re: Robust Data Gathering for Wireless Sensor Networks. In Proceedings of ICON (2005)
10. D. Tian and N. Georganas: Energy Efficient Routing with Guaranteed Delivery in Wireless Sensor Networks. In Proceedings of WCNC (2003)
11. J. Baek, Y. Nam, and D. Seo: ATS-DA: Adaptive Timeout Scheduling for Data Aggregation in Wireless Sensor Networks. In Proceedings of ICOIN (2007)
12. S. Li, Y. Lin, S. Son, J. Stankovic and Y Wei: Event Detection Services Using Data Service Middleware in Distributed Sensor Networks. Telecommunication Systems Vol.26 (2004)
13. N. Kurata, B. Spencer, M. Ruiz-Sandoval: Application of Wireless Sensor Network Mote for Building Risk Monitoring. In Proceedings of INSS (2004)
14. G. Werner-Allen, P. Wieskowski and M. Welsh: Demonstration: Real-Time Volcanic Earthquake Localization. In Proceeding of SenSys (2006)
15. A. Boukerche, R. Pazzi, and R. Araujo: A Fast and Reliable Protocol for Wireless Sensor Networks in Critical Condition Monitoring Applications. In Proceedings of MSWiM (2004)
16. VINT: The Network Simulator NS-2. http://www.isi.edu/nsnam, (2005)