

LOFT: A Latency-Oriented Fault Tolerant Transport Protocol for Wireless Sensor-Actuator Networks

Edith C.-H. Ngai Yangfan Zhou Michael R. Lyu
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Hong Kong, China
Email: {chngai, yfzhou, lyu}@cse.cuhk.edu.hk

Jiangchuan Liu
School of Computing Science
Simon Fraser University
Vancouver, BC, Canada
Email: jcliu@cs.sfu.ca

Abstract— Wireless sensor-actuator networks, or WSANs, refer to a group of sensors and actuators which collect data from the environment and perform application-specific actions in response. To act responsively and accurately, an efficient and reliable data transport protocol is crucial for the sensors to inform the actuators about the environmental events. Unfortunately, the low-power multi-hop communications in WSANs are inherently unreliable; the frequent sensor and link failures as well as the excessive delays due to congestion further aggravate the problem.

In this paper, we propose a latency-oriented fault tolerant data transport protocol in WSANs. We argue that reliable data transport in such a real-time system should resist to the transmission failures, and should also consider the importance and freshness of the reported data. We articulate this argument and provide a cross-layer two-step data transport protocol for on-time and fault tolerant data delivery from sensors to actuators. Our protocol adopts smart priority scheduling that differentiates the event data of non-uniform importance. It balances the workload of sensors by checking their queue utilization and copes with node and link failures by an adaptive replication algorithm. We evaluate our protocol through extensive simulations, and the results demonstrate that it achieves the desirable reliability for WSANs.

I. INTRODUCTION

Wireless sensor-actuator networks (WSANs), constructed by a group of sensors and actuators, have been suggested as a promising extension to traditional wireless sensor networks (WSNs) [1][2]. Actuators, which have much stronger computation and communication power, can process the data reported from the sensors and interact with the environment accordingly [3]. A well-designed communication module between the two types of nodes is crucial to WSANs [4]. In particular, given that the actuators need timely event data from the sensors to perform corresponding actions, reliability is an important concern in the sensor-actuator communication. Unfortunately, the low-power multi-hop communications in WSANs are inherently unreliable. The frequent sensor failures and the excessive delays due to node and link failures or congestion further aggravate the problem.

In this paper, we focus on the design of a latency-oriented fault tolerant data transport protocol in WSANs. We contend that the reliability in this context requires successful arrivals of packets within a latency bound, and a cross-layer design is thus necessary. Specifically, the queue utilization can serve as a good indication to the load of a node, and hence the delay

of an incoming packet. This can then be jointly optimized with the path success rate of transmission to handle node and link failures. We also suggest that the non-uniform importance of the events of interest can be explored in the optimization. We consequently present an importance-aware reliability index for the WSANs. Our protocol consists of two steps: First, it estimates the load of the neighbors and decides the next hops that can provide on-time delivery of event data to the actuator; and second, it copes with the transmission failures by providing redundant packets adaptively. Simulation results show that it remarkably improves the reliability for data reporting from sensors to actuators with reasonable overhead.

The remainder of this paper is organized as follows: Section II presents the related work. In Section III, we describe our design objective. The reliable data transport protocol is presented in Section IV, together with detailed descriptions of two designated steps, i.e., estimating the load of neighbors and coping with transmission failures. In Section V, we provide simulation results for our protocol. Finally, we conclude the paper in Section VI.

II. RELATED WORK

Efficient and reliable data transport is an important issue in WSNs and WSANs. He et al. [5] proposed a real-time communication protocol SPEED, which combines feedback control and non-deterministic QoS-aware geographic forwarding. Lu et al. [6] described a packet scheduling policy, called Velocity Monotonic Scheduling, which inherently accounts for both time and distance constraints. Felemban et al. [7] proposed Multi-path and Multi-Speed Routing Protocol (MMSPEED) for probabilistic QoS guarantee in WSNs. For reliable transmission with error-prone sensors, Aidemark et al. [8] presented a framework for achieving node-level fault tolerance (NLFT). It describes a lightweight NLFT approach that masks transient faults locally by using time-redundant task scheduling in the nodes.

Transmission failures frequently occur in wireless communications [9][10]. Dubois-Ferriere et al. [11] introduced a scheme for error-correction that exploits temporal and spatial diversity through packet combining. Ganesan et al. [12] described the use of multipath routing for energy-efficient recovery from node failures in wireless sensor networks, which

presents and evaluates the classical node-disjoint multipath and the braided multipath designs. S. Jain et al. [13] considered the problem of routing in a delay tolerant network in the presence of path failures. It improves the probability of successful packet delivery by applying a combination of erasure coding and data replication. Wang et al. [14] investigated direct transmission and flooding on the delay and fault tolerant mobile sensor network (DFT-MSN), and introduced an optimized flooding scheme that minimizes the transmission overhead of flooding.

Our work is motivated by the above studies. The key difference is that we provide an effective protocol for latency-oriented fault tolerant data reporting, and focus on the timely and reliable interactions between sensors and actuators. Although there have been studies exploring the heterogeneous sensor networks, e.g., [15][16], as well as wireless sensor-actuator networks [17], the reliability issues, in particular those for data transport from sensors to actuators, have yet to be addressed.

III. DESIGN OBJECTIVE

We consider a sensor-actuator network in which the sensors are responsible for collecting event data and a subset of them are responsible for forwarding the aggregated event data to the actuators for further actions. The communications from the sensors to the actuators follow an anycast paradigm; that is, a data transport is successful if any of the actuators receives the report. Similar to other geographic routing algorithms, the sensors and actuators are able to determine their coordinates by means of a location system like GPS or relative positioning based on signal strength estimation. Sensors also exchange location information with neighbors by periodic beaconing [5][7][18]. In addition, they make use of the beacons for estimating the workload of the neighbors in our protocol.

We propose a reliability index, which measures the probability that the event data are transmitted to the actuator successfully within a pre-defined latency bound. Each event also has an application-specific importance level in between 0 and 1. An event with higher importance is expected to achieve higher reliability. To realize this, each sensor maintains a priority queue, and, during transmission, important event data are scheduled with higher priorities. Moreover, replication is applied adaptively depending on the event importance and the link reliability to cope with transmission failures.

We give a formal description of the system parameters as shown in Table I. Our objective is to maximize the overall reliability index, \mathbb{R} , across all the events, as follows:

Maximize

$$\mathbb{R} = \sum_{\forall e} \left(\frac{Imp(e)}{\sigma} * r_e \right), \quad (1)$$

where $r_e = \frac{|Q_e|}{N_e}$ and $\sigma = \sum_{\forall e} Imp(e)$.

TABLE I
SYSTEM PARAMETERS

Event	e
Data report of event e	q_e
Set of data reports of event e that reach the actuator within the latency constraint	Q_e
Importance of event	$Imp(e)$
Latency bound for sensor-actuator reporting of event e	B_e
End-to-end delay of data report q_e	D_{q_e}
Number of data reports for event e	N_e

Subject to

$$D_{q_e} \leq B_e. \quad (2)$$

Clearly, the overall reliability of the system, \mathbb{R} , depends on the importance of the events and their respective reliability, r_e . The latter further depends on the reports reaching an actuator successfully within the latency bound.

IV. LATENCY-ORIENTED FAULT TOLERANT DATA TRANSPORT PROTOCOL

Our latency-oriented fault tolerant (LOFT) protocol addresses the problem of data transport from the event source to the actuator. It seamlessly integrates modules across different layers in achieving the above reliability objective. In this section, we first discuss the latency-oriented and importance-aware transmission through prioritized scheduling and routing for each sensor. Then, we provide a feedback algorithm to estimate the link qualities and determine the replication factor adaptively in the presence of node and link failures. It ensures a balanced, latency-oriented, and fault tolerant data transport process across different events in the network.

A. Estimating the Load of Neighbors

The key design objective here is to maximize the number of reports reaching the destination within their latency bounds, and, for different event types, to give preference to important events. Estimating the load of the neighbors allows a packet to be forwarded to a next hop with less queuing and transmission time. To this end, we adopt a priority queue in each sensor, which plays two important roles: 1) prioritized scheduling to speed up important event data transmission; and 2) queue utilization as an index for route selection to meet the latency bounds.

In our preemptive priority queue, packets for the event data are placed according to their data importance, and each priority queue is served in a first-in-first-out (FIFO) discipline. Our protocol is able to handle a network with heavy traffic, in which queuing delay can be a dominating factor over the processing and propagation delays.

Figure 1 shows how node i forwards packets to its neighbors j_1 , j_2 , and j_3 . The geographical distances from j_1 , j_2 , and j_3 to actuator a are represented by $\|j_1, a\|$, $\|j_2, a\|$, and $\|j_3, a\|$, respectively. Only the neighbors, which provide satisfactory advancement from i to a , will be considered as the next hop. Furthermore, the queue utilization of the neighbors is

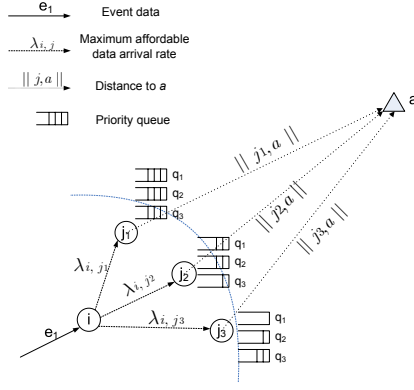


Fig. 1. Maximum affordable arrival rate from i to j .

considered in route selection. For example, the data e_1 flowing into i has the highest priority, so it will be served by the highest priority queue q_1 . Among all the neighbors of i , j_3 is selected as it provides e_1 with the best service by an empty highest priority queue q_1 .

The queueing delay of the highest priority queue is $\bar{d}_{q_1} = \bar{R} + \bar{S}N_{q_1}$, where $\bar{R} = \frac{1}{2} \sum_{k=1}^K \lambda_k \bar{S}^2$ is the mean residual service time in the node, N_{q_1} is the mean number of packets in first queue, K is the number of priority queues, λ_k is the arrival rate of the packets in priority queue k , and \bar{S} and \bar{S}^2 are respectively the expectation and second moment of the service time of the sensor. We assume the packet arrival is a Poisson process. \bar{S} can be obtained in each individual sensor by observing the time it takes to serve a packet.

More formally, consider node i that receives a new event data $data_e$. It obtains control packets from its neighbors j , $\langle a, \bar{S}, \lambda_{high}, \lambda_{low} \rangle_j$, where a is the target actuator, \bar{S} is the expected service time of node j , $\lambda_{high} = \sum_{\forall k, Imp(data_k) \geq Imp(data_e)} \lambda_k$ is the sum of all arrival rates λ_k of the data that are equal to or more important than $data_e$, and $\lambda_{low} = \sum_{\forall k, Imp(data_k) < Imp(data_e)} \lambda_k$ is the sum of all λ_k of the data that are less important than $data_e$.

Node i needs to ensure that the end-to-end latency for $data_e$ is no more than the latency bound B_e . To this end, it first estimates the advancement $h_{i,j}$ towards the actuator a from i to j , and then the maximum hop-to-hop delay from i to j , $delay_{i,j}$. Note

$$h_{i,j} = \frac{\|a, i\| - \|a, j\|}{\|a, i\|}. \quad (3)$$

So,

$$delay_{i,j} \leq B_e * h_{i,j}. \quad (4)$$

Since $delay_{i,j} = d_q + d_{tran} + d_{prop} + d_{proc}$, the maximum queueing delay $d_{q_{max}}$ is:

$$d_{q_{max}} = B_e * h_{i,j} - (d_{tran} + d_{prop} + d_{proc}). \quad (5)$$

Only neighbors with $d_{q_{max}} > 0$ will be considered as the

next hop; otherwise, the latency bound cannot be met. Among these candidates, node i starts inspecting the neighbors with both $\lambda_{low} = 0$ and $\lambda_{high} = 0$, followed by the remaining neighbors. For each candidate above, node i calculates the maximum data rate λ_i that it can forward while satisfying the latency bound:

$$d_{q_{max}} > \frac{\bar{R}}{(1 - \lambda_{high}\bar{S})(1 - \lambda_{high}\bar{S} - \rho_{i,j})}, \quad (6)$$

and

$$\rho_{i,j} < 1 - \lambda_{high}\bar{S} - \frac{\bar{R}}{(1 - \lambda_{high}\bar{S})d_{q_{max}}}, \quad (7)$$

where $\rho_{i,j} = \lambda_{i,j}\bar{S}$ is the maximum affordable load of j for handling data from i on event e .

Then the event data packets are forwarded to the neighbor with the highest $h_{i,j}$ and satisfactory $\lambda_{i,j}$, which is the closest to the destination with enough capacity for transmission. Each intermediate node updates the latency bound B_e before forwarding the packet to the next hop.

B. Coping with Transmission Failures

As mentioned before, packets will be dropped if they expire before reaching the actuators. Apart from that, data may be lost due to link failures, such as link transmission errors, buffer overflow, or node failures along the path. However, there exist multiple destinations (actuators) and multiple paths for anycast data transport in WSANs. Different levels of reliability can therefore be obtained based on the requirements of various event data. We adopt adaptive packet replications to handle link failures and provide reliability in terms of the success arrival of packets. In this section, we extend the above routing algorithm to cope with transmission failures in data routing.

For simplicity, we consider that the event reliability requirement R_{req} is proportional to its event importance. For example, an event with important level of 0.8 will have the reliability requirement of 0.8. We define link loss rate $L_{i,j}$ as the packet loss rate from node i to its next hop j , and path success rate P_j as the probability that a packet from node j reaches the actuator (destination) successfully. Instead of forwarding a packet to one next hop with the highest $h_{i,j}$ and satisfactory $\lambda_{i,j}$, node i forwards the packet to multiple next hops and decides the replication factor r_f of packets adaptively.

Again, consider node i and its potential next hops j_1, j_2 , and j_3 in Figure 2. The observed link loss rate from i to j_1, j_2 , and j_3 are L_{i,j_1}, L_{i,j_2} , and L_{i,j_3} . Based on these link loss rates, the corresponding path success rates P_{j_1}, P_{j_2} , and P_{j_3} from j_1, j_2 , and j_3 to a are estimated. The allocation of packets from i to its neighbors is proportional to their maximum affordable arrival rates $\lambda_{i,j_1}, \lambda_{i,j_2}$, and λ_{i,j_3} to balance the load. After that, node i may check if the estimated path success rate can meet the event reliability requirement R_{req} . If not, it decides the replication factor r_f to meet the requirement and forwards the replicated packets to the next hops.

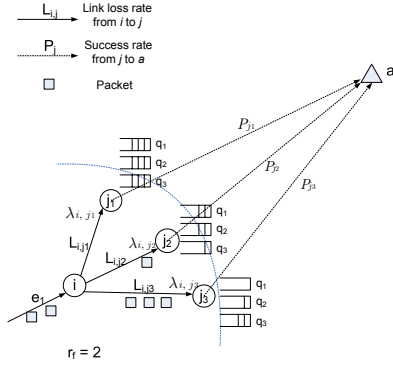


Fig. 2. Forwarding packets with replication factor $r_f=2$.

We now discuss the above process in detail. First, node i selects the top k neighbors with the highest $h_{i,j}$ and satisfactory $\lambda_{i,j}$, and estimates their link loss rates $L_{i,j}$. Each neighbor j periodically reports the number of packets it received from node i , so that i can calculate the loss rate $L_{i,j}$ with the number of packets it sent to j in a particular time interval. Then, it can obtain the link loss rate by an EWMA (Exponentially Weighted Moving Average) [19] approach with its previous and current estimations of the link loss rate. Then, i estimates the path success rate P_j from i to a via j as follow:

$$P_j = (1 - L_{i,j})^{1/h_{i,j}}. \quad (8)$$

Sensor i will allocate the packets to its neighbors according to their $\lambda_{i,j}$. The neighbors with higher $\lambda_{i,j}$ will be allocated with more code blocks. The proportion $prop_j$ of packets to neighbor j is:

$$prop_j = \frac{\lambda_{i,j}}{\sum_{n=1}^k \lambda_{i,n}}. \quad (9)$$

The probability that the packet can be delivered successfully from i to a by these k neighbors, P_i , can then be estimated as:

$$P_i = \sum_{j=1}^k \left(\frac{\lambda_{i,j}}{\sum_{n=1}^k \lambda_{i,n}} * P_j \right). \quad (10)$$

Then, node i determines the replication factor r_f with the following equation:

$$r_f = ceil(R_{req}/P_i). \quad (11)$$

The replication factor r_f must be greater than R_{req}/P_i , where R_{req} is initialized as the required event reliability, or the event importance in our work, by the event source. Each of the neighbors above will be allocated with proportion $prop_j$ of packets from i . The corresponding path success rate P_j will become the required reliability R_{req} of that particular path from j to the actuator.

Each node j , which received the packets, selects the next hop m' with the highest $h_{j,m'}$ and satisfactory $\lambda_{j,m'}$. Similarly, the path success rate obtained must be greater than R_{req} :

$$(1 - L_{j,m'})^{1/h_{j,m'}} \geq R_{req}. \quad (12)$$

If the link loss rate from j to m' satisfies the above equation, packet will be forwarded to m' . Since the reliability of a path is composed by a series of links on it:

$$(1 - \bar{L}_1)(1 - \bar{L}_2)(1 - \bar{L}_3)...(1 - \bar{L}_n) > R_{req}, \quad (13)$$

and

$$(1 - \bar{L}_2)(1 - \bar{L}_3)...(1 - \bar{L}_n) > R_{req}/(1 - \bar{L}_1), \quad (14)$$

where the $\bar{L}_1, \bar{L}_2, \dots, \bar{L}_n$ are the packet loss rates of the links on the path.

Node j updates the reliability R_{req} and forwards it with the packets to the selected neighbor m' :

$$R'_{req} = R_{req}/(1 - L_{j,m'}). \quad (15)$$

In case that $\bar{L}_{j,m'}$ does not satisfy the required reliability, node j will look for the neighbor with the next highest $h_{j,m}$ and satisfactory $\lambda_{j,m}$. The process is repeated until it goes through all the potential neighbors with high $h_{j,m}$ and $\lambda_{j,m}$. If no single neighbor can provide low enough link failure rate, j forwards packets to multiple neighbors and decides the replication factor, as shown in node i .

V. EVALUATION

We have conducted simulations for our proposed reliable data transport protocol in NS-2 network simulator [20]. The simulation settings are mainly drawn from [5], which are summarized in Table II. We consider a loss model in which packet loss occurs because of poor channel quality [13]. Every link is characterized by a failure probability, which is the probability that a packet is dropped during the transmission to the next hop.

TABLE II
SIMULATION PARAMETERS

Network size	200m x 200m
No. of sensors	100
Node placement	Uniform
Radio range	40m
MAC layer	IEEE 802.11
Bandwidth	2 Mbps
Packet size	32 bytes
Control packet rate	1 pkt/s
B_e	2 sec

We evaluate the event reliability, average delay of individual events, and overall reliability obtained in our latency-oriented fault tolerant (LOFT) data transport protocol. Two events are generated randomly in the network with the event importance 1.0 and 0.4, respectively. Each packet should be reported to the

actuator within the latency bound of 2 sec. For comparison, we also show the results of the priority-based event reporting with event importance approach (PREI) [21], where priority-based forwarding is employed, but the link failures are not considered. We repeat the experiments for 10 times by changing the random seed and show the average results.

A. Protocol Performance with Varying Link Failure Probability

In this experiment, we fix the data rate at 15 pkt/s and vary the link failure probability f . It means that there is a probability f for each link to encounter transmission failure when forwarding a packet to the next hop. Figure 3(a) shows that LOFT protocol achieves much higher event reliability than PREI approach for both types of events. The reliability of the more important event (event 1) is higher than that of the less important event (event 2) in LOFT. This follows our design objective that important events should be guaranteed with higher reliability. On the contrary, the reliability of the two events are similar in the PREI, though priority-based routing is applied. It is because PREI has no mechanism to handle link failures, and hence cannot provide any differentiation on the reliability among different events.

Figure 3(b) further shows the average delay in LOFT and PREI. PREI performs a bit better than LOFT when the link failure probability is low. This is because it always selects the next hop with the lightest workload, while LOFT also estimates and considers the link packet loss rates when selecting the route. However, it is clear that the delay in LOFT is lower than that in PREI when the link failure probability increases. The reason is that replication is applied in LOFT, so packets are routed through multiple paths. Intuitively, it achieves a lower data delivery delay.

The overall reliability index, \mathbb{R} , of the two protocols is shown in Figure 3(c). It demonstrates that LOFT outperforms PREI, and the gap increases when the link failure probability becomes higher.

B. Protocol Performance with Varying Data Rate

We further study the effect of data rates to the performance of our protocol. We fix the link failure probability as 0.05 and vary the data rates. Figure 4(a) shows that our LOFT protocol can achieve nearly perfect reliability, while PREI can only achieve reliability close to 0.8. It also indicates that the reliability achieved is independent of the data rates. Similarly, Figure 4(b) shows that LOFT achieves small and comparable average delay with PREI. Note that, the average delay of the less important event (event 2) in LOFT increases with the data rates. It is because the traffic load of the network increases with replication under a high data rate. The queuing and transmission times may then become non-negligible for the low-priority packets. Figure 4(c) again shows that the overall reliability of LOFT is higher than that of PREI.

C. Protocol Overhead

We discuss the overhead of the PREI and LOFT protocol here. There are basically three types of overhead, including

1) location update packets, 2) control packets for estimating the loads and link loss rates of neighbors, and 3) replicated data packets for handling link failures. Since the location update packets are inherited from traditional geographic routing protocols [5][7][18], we focus on the remaining two types of overhead.

The control packets for estimating the load of neighbors are required for both PREI and LOFT. LOFT further utilizes these control packets for estimating the link loss rate of neighbors, so no extra packets are required. In our experiment, the control packet rate is 1 pkt/s, which is relatively low in comparison with the data rate. Regarding the replicated data packets, we evaluate the data replication factor in LOFT. Figure 5(a) shows that the replication factor is around 2 when the link failure probability is between 0 and 0.3 with data rate 15 pkt/s. The replication factor increases with the link failure probability to cope with the packet loss due to transmission failures. Similarly, Figure 5(b) also shows that the replication factor is independent of the data rates. The above results indicate that our protocol overhead is affordable in meeting the reliability objective for latency-oriented and fault tolerant data transmission.

VI. CONCLUSION

In this paper, we proposed a reliable data transport protocol in wireless sensor-actuator networks (WSANs). We considered that the system reliability in this context is closely related to the delay and the resistance to link failures, which should be jointly optimized. We also suggested that the non-uniform importance of the event data can be explored in the optimization procedure. Following these arguments, we proposed a general latency-oriented fault tolerant data transport protocol. It adopts smart priority scheduling and applies replication of packets adaptively in handling link failures. The protocol consists of two steps. It first locates a list of neighbors that provide satisfactory advancement to the actuator by checking their locations, and then estimates their maximum affordable incoming data rates through analyzing their priority queue utilization. To cope with transmission failures, packet loss rates of the links are updated regularly. Packets are allocated to the next hops in proportion to their maximum affordable data rates. Replication of packets is applied adaptively based on the required data reliability and the estimated path success rates for delivery. The performance of the proposed protocol has been examined through extensive simulations. The results demonstrated that our protocol can significantly enhance the reliability for data transport, while the average delay for high priority packets is minimized.

ACKNOWLEDGMENT

The work described in this paper was substantially supported by grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4205/04E). J. Liu's work was supported in part by a Canadian NSERC Discovery Grant 288325, an NSERC Research Tools and Instruments Grant, a Canada Foundation for Innovation (CFI) New Opportunities Grant, and an SFU President's Research Grant.

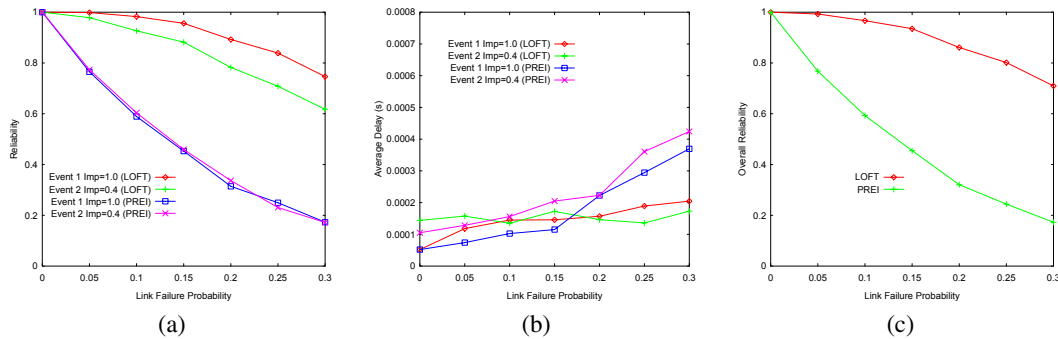


Fig. 3. Protocol performance on (a) event reliability (b) average delay (c) overall reliability with data rate 15 pkt/s.

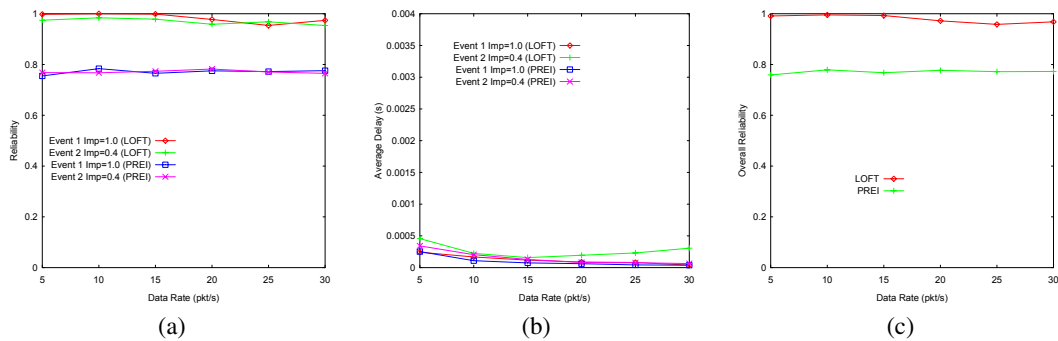


Fig. 4. Protocol performance on (a) event reliability (b) average delay (c) overall reliability with link failure probability 0.05.

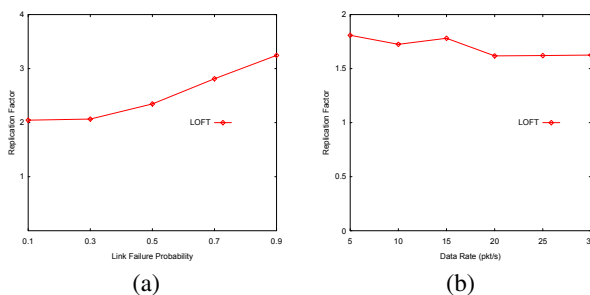


Fig. 5. Replication factor with (a) data rate 15 pkt/s (b) link failure probability 0.05.

REFERENCES

- I. F. Akyildiz, W. Su, and T. Sandarasubramaniam, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 5, pp. 393–422, 2002.
- D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," in *Proc. of ACM MobiCom*, Seattle, Washington, U.S., 1999.
- L. Girod, N. Ramanathan, J. Elson, T. Stathopoulos, M. Lukac, and D. Estrin, "Emstar: a software environment for developing and deploying heterogeneous sensor actuator networks," *ACM Transactions on Sensor Networks*, 2007.
- I. F. Akyildiz and I. Kasimoglu, "Wireless sensor and actor networks: research challenges," *Elsevier Ad Hoc Networks Journal*, Oct 2004.
- T. He, J. Stankovic, C. Lu, and T. Abdelzahr, "SPEED: a real-time routing protocol for sensor networks," in *Proc. of the IEEE ICDCS*, Providence, RI, U.S., May 2003, pp. 46–55.
- C. Lu, B. M. Blum, T. F. Abdelzahr, J. A. Stankovic, and T. He, "RAP: a real-time communication architecture for large-scale wireless sensor networks," in *Proc. of the IEEE RTAS*, San Jose, CA, U.S., Sep 2002.
- E. Felemban, C.-G. Lee, E. Ekici, R. Boder, and S. Vural, "Probabilistic QoS guarantee in reliability and timeliness domains in wireless sensor networks," in *Proc. of the IEEE Infocom*, Miami, FL, U.S., Mar 2005.
- J. Aidemark, P. Folkesson, and J. Karlsson, "A framework for node-level fault tolerance in distributed real-time systems," in *Proc. of the IEEE DSN*, Yokohama, Japan, Jun 28 - Jul 1, 2005.
- J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," in *ACM Sensys*, Nov 2003.
- A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multiple routing in sensor networks," in *ACM Sensys*, Nov 2003.
- H. Dubois-Ferriere, D. Estrin, and M. Vetterli, "Packet combining in sensor networks," in *ACM Sensys*, San Diego, U.S., Nov 2005.
- 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. 1, no. 2, 2001.
- S. Jain, M. Demmer, R. Patra, and K. Fall, "Using redundancy to cope with failures in delay tolerant network," in *Proc. of the ACM SIGCOMM*, Pennsylvania, U.S., Aug 2005.
- Y. Wang and H. Wu, "DFT-MSN: The delay fault tolerant mobile sensor network for pervasive information gathering," in *Proc. of the IEEE Infocom*, 2006.
- V. P. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff, "A minimum cost heterogeneous sensor network with a lifetime constraint," *IEEE Transaction on Mobile Computing*, vol. 4, no. 1, Jan/Feb 2005.
- M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh, "Exploiting heterogeneity in sensor networks," in *Proc. of the IEEE Infocom*, Miami, FL, U.S., Mar 2005.
- T. Melodia, D. Pompili, V. C. Gungor, and I. F. Akyildiz, "A distributed coordination framework for wireless sensor and actor networks," in *Proc. of ACM Mobihoc*, Urbana-Champaign, IL, U.S., 2005, pp. 99–110.
- H. Frey and I. Stojmenovic, "On delivery guarantees of face and combined greedy-face routing in ad hoc and sensor networks," in *Proc. of the ACM MobiCom*, Sep 2006.
- C. A. Lowry, W. H. Woodall, C. W. Champ, and S. E. Rigdon, "A multivariate exponentially weighted moving average chart," *Technometrics*, vol. 34, pp. 46–53, 1992.
- K. Fall and K. Varadhan, *The ns manual*, Dec 2003, <http://www.isi.edu/nsnam/ns>.
- E. C.-H. Ngai, Y. Zhou, M. R. Lyu, and J. Liu, "Reliable reporting of delay-sensitive events in wireless sensor-actuator networks," *The Third IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'06)*, Oct 2006.