Designing a Sensor Network Architecture for Transmission of Urgent Sensor Information

Tetsuya Kawai, Naoki Wakamiya, and Masayuki Murata
Graduate School of Information Science and Technology
Osaka University, Osaka, Japan
Email: {t-kawai, wakamiya, murata}@ist.osaka-u.ac.jp

Abstract— **In wireless sensor networks used as a social infrastructure, urgent information, such as a fire alarm, is needed to be transmitted as fast and reliably as possible. In this paper, we propose design methodology for a sensor network which provides preferential control of urgent information over other non-urgent information. In our methodology, several simple mechanisms which function in different spatial and temporal levels are introduced on each node and they work autonomously and independently as a reaction to the surrounding situation. We also show an example of a network architecture designed following the methodology. Our simulation experiments showed that the architecture successfully improved the delivery ratio and delay of the urgent sensor information.**

I. Introduction

Wireless Sensor Network (WSN) technology is expected to play an essential role for our society in the near future. A WSN consists of a number of sensor nodes and a base station (BS). A node is equipped with a processing unit, a radio transceiver, and sensors. Nodes are deployed in a region to monitor and then environmental information detected by sensors is collected to a BS through wireless communication among sensor nodes [1].

WSN technology will be used for a wide variety of applications, such as agricultural, health, environmental, and industrial purposes. Among them, a WSN used as a social infrastructure to make our life safe, secure, and comfortable is one of the most promising. This sort of WSNs is supposed to carry various types of information, such as temperature, humidity, fire alarm, intrusion warning, seismic movement, image, and sound. The urgent information, a fire alarm for example, has to be transmitted through a WSN with higher reliability and lower latency than other non-urgent information.

There have been several proposals of QoS (Quality of Service) control for WSNs [2], [3]. ESRT [4] achieves the desired reliability by regulating the packet emission rate at source nodes with feedback from a BS. In [5], overheard packets are used for error correction to improve the reliability in both of single-hop and multihop communications. In [6], the authors propose a routing protocol to find the best path in terms of delay. For congestion mitigation in a WSN, CODA [7] combines hop-by-hop backpressure and end-to-end feedback techniques. In [8], a prioritized MAC protocol is introduced in addition to a hop-by-hop traffic control. IFRC [9] is developed to realise adaptive fair and efficient rate allocation by sharing information on the level of congestion among nodes.

The main contribution of this paper is proposal of design methodology of a network architecture for transmission of urgent sensor information. In comparison to the research works mentioned above, our approach is unique in that several simple mechanisms are incorporated above the network layer. In order to adapt to the scale of an emergency ranging from a small event like a gas leakage to a catastrophic event such as an earthquake attack, some simple mechanisms are embedded in each node, instead of developing a monolithic and complicated mechanism. Those mechanisms work in different spatial and temporal levels and they autonomously and independently react to its surrounding situation locally observed. When a small event happens, it is not necessary to involve all nodes to respond to the emergency. Instead, only nodes along the path from the node which detects the event to a BS participate in transmission of urgent information and adopt a hop-by-hop retransmission mechanism for example. As the scale of the emergency grows over time, additional mechanisms come into effect and more nodes become involved in the urgent information transmission. On the other hand, in the event of a large emergency, a lot of nodes detect the emergency at the same time and send urgent information simultaneously and independently from others. A WSN in this case immediately reacts as a whole to control serious congestion for fast and reliable transmission of urgent information. In this paper, we also show a network architecture called UMIUSI (aUtonomous Mechanisms Integrated for Urgent Sensor Information), which incorporates five simple mechanisms following the above methodology.

The rest of this paper is organized as follows. Section II proposes design methodology for fast and reliable transmission of urgent information and Section III introduces our UMIUSI architecture. Evaluation of the architecture by simulation experiments is presented in Section IV. Finally, we conclude this paper in Section V.

II. Design Methodology

A. System Overview

In this paper, we consider a WSN deployed in a building or a house to monitor and control a living and working environment. A WSN consists of one BS and a number of immobile sensor nodes. Sensor nodes are operated on a battery and equipped with a variety of sensors. Normally, each node observes its environment and reports obtained sensor

1-4244-1557-8/07/$25.00 ©2007 IEEE

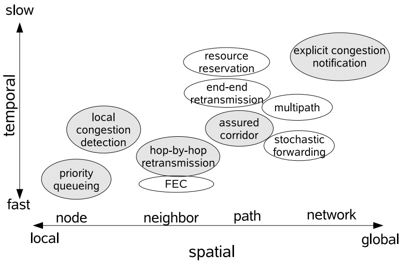

Fig. 1. Examples of control mechanisms.

information to the BS at regular intervals. A way that sensor information is transferred to the BS depends on a routing protocol or a data gathering mechanism employed. To save its power consumption, a sensor node sleeps and wakes up in accordance with a sleep scheduling algorithm. Once a node detects an emergency or an unusual condition, it begins to emit packets containing urgent information at shorter intervals.

Since the methodology incorporates mechanisms above the network layer, any MAC protocols, routing protocols, data gathering mechanisms, and sleep scheduling algorithms can be incorporated with a network architecture designed following the methodology, with no or small modification. However, as a base of consideration, we assume a contention-based MAC protocol and a multihop routing protocol in this paper. This assumption does not spoil the applicability of the proposal, because contention-based, *e.g.*, CSMA/CA-type of MAC protocols are preferred for its simplicity and multihop routing protocols are often used for energy saving and spatial reuse of wireless channels.

B. Design methodology

Since a large number of battery-driven nodes are deployed in a WSN, energy efficiency, fault tolerance, and scalability should be taken into account in designing a WSN architecture [1]. In addition, reliability and latency should be considered to satisfy application requirements. Especially in an event of emergency, reliability and latency of transmission of urgent information are crucial.

For operations under normal conditions, a WSN adopts a MAC protocol, a routing protocol, a data gathering scheme, and a sleep scheduling algorithm, which fit to application requirements. Once an emergency occurs, an appropriate series of actions should take place to deliver urgent information to a BS with satisfactory high reliability and low latency. Control should be local for a small-scale event keeping nodes which are not involved in the emergency operating as usual. On the other hand, as an event grows or a large-scale event occurs, a WSN should react as a whole for fast and reliable transmission of urgent information.

In summary, our design objectives of a WSN architecture for transmission of urgent sensor information are:

- *High reliability and low latency*. The reliability and latency of transmission of urgent information are the most important issues. Urgent information should be differentiated from other information and receive preferential controls according to their importance. We consider that energy efficiency can be sacrificed to some extent for transmission of urgent information.
- *Self-organizing and localized behavior*. The type and scale of an emergency and the number of simultaneous emergency events are unpredictable and dynamically change as time passes. A centralized architecture is infeasible in an emergency due to variations of traffic pattern and the level of congestion. Therefore, we need an architecture which is fully-distributed, self-organizing, and adaptive to dynamically changing conditions. As a consequence of localized reactions of each sensor node to the surroundings and local interactions among nodes, a globally-organized behavior of a WSN against detected emergencies emerges as a whole.
- *Simplicity*. Since a node has limited computational capacity and a small amount of memory, mechanisms to support fast and reliable transmission of urgent information must be simple enough. Simplicity also contributes to low energy consumption and less programming errors.

To satisfy the above requirements, we propose design methodology to combine several simple control mechanisms which function in different temporal and spatial levels. In Fig. 1, typical control mechanisms are arranged in accordance with their temporal and spatial effect. In general, larger the spatial area where a mechanism influences is, longer the time required to achieve effective control is. In the methodology, at least one mechanism is chosen for each of spatial levels.

The collective behavior of these mechanisms offers appropriate preferential transmission of emergency packets. At the beginning of an event, quick-acting node-level and neighbor-level mechanisms contribute to fast and reliable transmission until slower path-level mechanisms come into effect. As the event develops and situation becomes more serious, additional mechanisms eventually become effective and network-level control is conducted. In our UMIUSI architecture, which will be proposed in Section III, those five mechanisms indicated by gray circles are embedded in a sensor node.

We should note here that it is not possible to transmit all emergency packets with high reliability and low latency because the capacity of a WSN is limited. Therefore, it is necessary to classify sensor information into several classes in accordance with the required QoS in terms of delay and reliability. Classification and prioritization can be determined beforehand. Context-aware prioritization is also helpful to adapt to dynamically changing emergency conditions. Each packet has a field in its header to indicate its corresponding class and packets in different classes are treated in a different way in a WSN.

III. UMIUSI Architecture

As an example of a network architecture designed following our methodology, we propose UMIUSI (aUtonomous Mechanisms Integrated for Urgent Sensor Information).

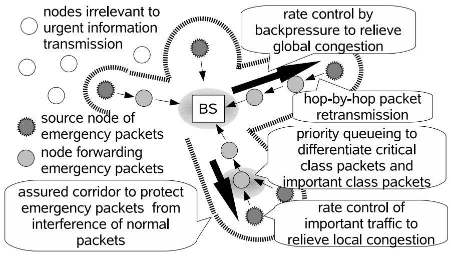
nodes irrelevant to urgent information transmission
rate control by backpressure to relieve global congestion
BS
hop-by-hop packet retransmission
source node of emergency packets
node forwarding emergency packets
priority queueing to differentiate critical class packets and important class packets
assured corridor to protect emergency packets from interference of normal packets
rate control of important traffic to relieve local congestion

Fig. 2. The mechanisms leveraged in UMIUSI.

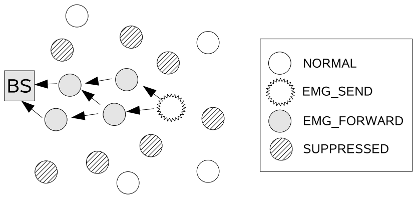

Fig. 3. An assured corridor.

A. Details of UMIUSI

We categorize sensor information into three classes as one normal class and two emergency classes. Emergency class information is prioritized over normal class information. The two types of urgent information are distinguished in more important and less important by the application layer.

- *Normal Class.* Any non-urgent information belongs to this class. Normal class information is gathered to a BS at regular intervals of t_{norm}. Without an emergency, reliability and latency of normal class information should satisfy application requirements by an adopted data gathering scheme. An application can tolerate delay and loss of normal class information under emergency conditions. Packets of this class are called normal packets.
- *Important Class.* This class is for urgent information, but an application can tolerate loss and delay of important class information to some extent. Packets belonging to this class, called important packets, can be delayed or dropped depending on the level of congestion under emergency condition. The interval of emission of important packets $t_{\mathrm{imp}} < t_{\mathrm{norm}}$ is determined by an application, but could be regulated to be larger than t_{norm} to mitigate congestion.
- *Critical Class.* This class is for the most urgent and important information which requires highly reliable and fast transmission to a BS. Critical packets are emitted by a node detecting an emergency at fixed regular intervals of $t_{\mathrm{cri}} < t_{\mathrm{norm}}$, which is determined by an application. The total amount of critical class traffic should not exceed the network capacity to guarantee a high delivery ratio and low delay of the required level. Therefore, the number of sensor nodes for critical information should be limited at the deployment, or some of them should be categorized into the important class. However, it is not a trivial task and remains as one of future works.

Following the methodology, we choose five mechanisms for UMIUSI: priority queueing, rate control by local congestion detection, retransmission, assured corridor mechanism (ACM) [10], and rate control by backpressure. These mechanisms are simple and distributed, work independently of each other, and cover all the levels from node-level to network-level. Figure 2 briefly summarizes how and where they work. Although another combination with other mechanisms is also possible, we must carefully consider the relation among mechanisms. For example, we can also introduce in-network aggregation [11] for reducing the number of packets and thus suppressing congestion occurrence. However, adopting both hop-by-hop and end-to-end retransmission mechanisms is likely to be inefficient or redundant.

The detailed description of each mechanism is presented in the following.

1) Assured corridor mechanism (ACM): The main purpose of this mechanism is to avoid loss of emergency packets, *i.e.*, packets of critical or important class, caused by collisions with normal packets. In addition, ACM contributes to avoiding delay caused by sleeping nodes. ACM establishes an “assured corridor” from a source node to a BS, in which emergency packets are protected from normal packets. An example of an assured corridor is illustrated in Fig. 3.

An assured corridor consists of awake nodes, which is on the path from a source node to a BS, and surrounding silent nodes, which are in the range of radio signals of the awake nodes. In normal operation, all nodes are in the *NORMAL* state and operate in accordance with a data gathering scheme. Once a node detects an emergency, it moves to the *EMG_SEND* state and begins to periodically emit critical or important packets. On receiving an emergency packet for the first time, other nodes move to either of the *SUPPRESSED* or *EMG_FORWARD* state. A node on the path to the BS is responsible for forwarding emergency packets to the BS. Therefore, it moves to the *EMG_FORWARD* state, cancels its sleep schedule to keep awake, and immediately relays emergency packets it receives. A node which receives an emergency packet but is not on the path moves to the *SUPPRESSED* state. A node in the *SUPPRESSED* state completely stops sending normal packets or decreases the sending rate of normal packets. When it detects a new emergency by itself or it receives an emergency packet as a relaying node, it also moves to either of the *EMG_SEND* or *EMG_FORWARD* state. Details of ACM with simulation results can be found in [10].

2) Retransmission: In order to recover a lost emergency packet while providing differentiated services, we introduce a prioritized scheduling algorithm to hop-by-hop retransmission. The algorithm can be incorporated with most retransmission mechanisms in MAC or above layer, such as that in [12].

A node retransmits an emergency packet when it detects a loss. The hop-by-hop acknowledgement can be easily done by, for example, overhearing a packet sent by a next-hop node. If the overheard packet does not contain the information that the node sent, the packet is considered to be lost. In this case, the

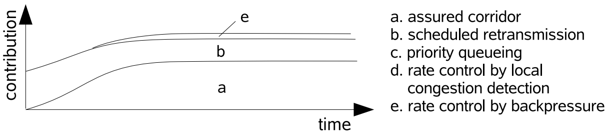

Fig. 4. The contribution of each mechanism for a small-scale event.

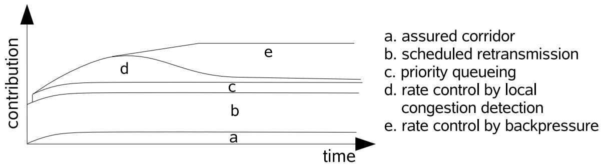

Fig. 5. The contribution of each mechanism for a large-scale event.

first retransmission is scheduled after a backoff. To prioritize retransmission of a critical packet, a backoff timer for a critical packet is set shorter than that for an important packet. If the first retransmission fails, one or more trials are conducted by applying doubled backoff, *i.e.*, a binary backoff scheme, until retransmission succeeds or a next-hop node goes to sleep.

3) Priority queueing: Each node has a priority queue for emergency packets, with which important packets are served only when there is no critical packet queued. This means that fast transmission of critical packets is accomplished at the sacrifice of longer transmission delay of important packets. Transmission of normal packets at a node during an emergency is delayed until the node returns to normal operation.

4) Rate control by local congestion detection: To mitigate congestion as fast as possible by local control, we introduce a rate control mechanism which is triggered by detection of local congestion. In order to keep the reporting rate of critical information at $1/t_{\mathrm{cri}}$, the rate control is applied only to important class traffic. When a source node of important packets detects local congestion, it increases the emission interval of important packets. Congestion detection can be done by, for example, observation of queue occupancy and channel sampling such as proposed in [7], [8]. As a rate control algorithm, a TCP-like AIMD (Additive Increase and Multiplicative Decrease) algorithm, such as that in [13], is empirically employed in our simulation experiments.

5) Rate control by backpressure: In an event of large emergency such as an earthquake, even if emission of normal packets is suppressed and source nodes of important packets regulate their emission rate, congestion cannot be fully avoided around a node belonging to multiple paths and around a BS, where many emergency packets concentrate on. We employ a backpressure mechanism for a network-level traffic control in UMIUSI. When a node detects congestion, it sets an explicit congestion notification (ECN) bit in the header of important packets which it relays toward a BS. By overhearing the packet, a preceding node, *i.e.*, a node closer to the source, recognizes that congestion occurs in the path to the BS. Then, it also sets an ECN bit of the next important packet which it relays to the next-hop node. Consequently, by means of overhearing, a congestion notification propagates to the source node. On receiving the notification, the source node reduces the emission rate of important packets, and the congestion is mitigated.

B. Discussion

As stated above, UMIUSI is designed to be able to adapt to a variety of emergencies by combining the five autonomous mechanisms. Whereas there is neither a mechanism to identify the type and scale of an emergency nor an explicit rule to choose and coordinate mechanisms, and these mechanisms work independently, resultant effects of the mechanisms working in different spatial and temporal levels give appropriate traffic control as a whole regardless of the phase and scale of an emergency. We choose the five mechanisms of UMIUSI so that the mechanisms do not conflict with but complement each other. For example, quick-acting node- or neighbor-level mechanisms complement slow path- or network-level mechanisms at the initial phase of an event. Figures 4 and 5 are intuitive sketches to show how the mechanisms collaborate by depicting the contribution of each mechanism to fast and reliable transmission of urgent information in a small-scale and large-scale event respectively.

In a small-scale event, only one or a few nodes would detect it. At the beginning of the event, the retransmission mechanism becomes effective immediately since an emergency packet is not protected from normal packets until an assured corridor is established. Once a corridor is established, contribution of the retransmission mechanism becomes smaller as the number of packet loss is reduced by ACM. The priority queueing mechanism is not used since all emergency packets are likely to belong to one class in a small-scale event. In addition, rate control of important class traffic by local congestion detection does not help much, since the number of nodes emitting important packets is small and the possibility of congestion is expected to be small. If multiple paths are established from a source node to a BS, there may occur collisions among emergency packets traversing different paths. In such a case, the rate control mechanism by backpressure is activated. Figure 4 is an intuitive sketch to show how much each mechanism contributes to fast and reliable transmission of emergency packets in a small-scale event.

In a large-scale event, since most of nodes are involved in transmission of emergency packets as source nodes or forwarding nodes, an assured corridor to suppress the emission of normal packet does not help much. On the contrary, mechanisms to mitigate congestion within a corridor are effective. The priority queueing mechanism offers differentiated forwarding services to emergency packets in accordance with their class. Rate control is first applied locally at a source node to mitigate local congestion among neighboring source nodes. Afterwards, congestion among emergency packets traversing different paths is solved by the backpressure mechanism.

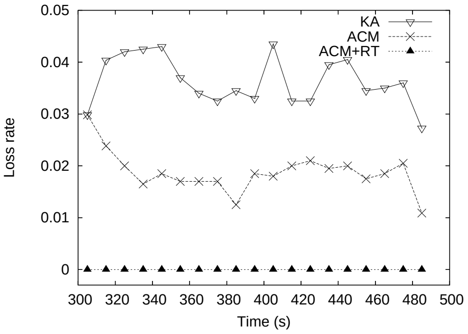

Fig. 6. The loss rate of emergency packets (small-scale event).

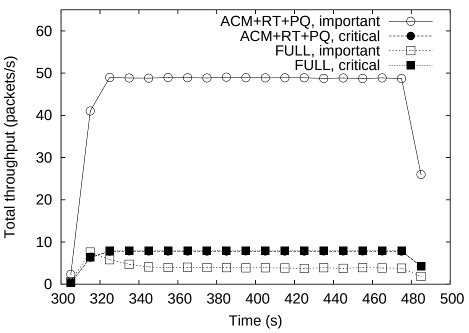

Fig. 7. The total throughput of important packets (large-scale event).

IV. Simulation Experiments

A. Simulation model

We implemented UMIUSI for the ns-2 network simulator package and conducted extensive simulation experiments. In all of the simulation experiments, 200 sensor nodes are uniformly and randomly distributed in a 20 m × 20 m two-dimensional region with a BS at its center. IEEE 802.15.4 non-beacon mode is used as the MAC protocol and the transmission range of radio signals is set to 2.5 m.

We employ the synchronization-based data gathering scheme [14] for the underlying data gathering mechanism. It employs a broadcast-based routing, and timing of packet emission is the same among nodes of the same hop distance from a BS. In a normal state, all nodes adopt a sleep schedule. Nodes on the same hop distance wake up at the same time and receive packets from one-hop distant node. Then, they send packets to next-hop nodes. Finally, after overhearing packets emitted by the next-hop nodes, they go back to a sleep mode. In the simulation experiments, we set the interval of normal packet emission t_{norm} at ten seconds, and the offset between emissions of adjacent nodes at one second. It means that a node is awake for two seconds in a data gathering interval of ten seconds.

Each simulation experiment lasts for 500 seconds including 300 seconds for initialization without any emergency. One (small-scale event) or 32 (large-scale event) nodes are randomly chosen after the initialization, and they detect an event at randomly chosen time within following 10 seconds. They wait for next-hop nodes to wake up, then start transmitting emergency packets at intervals of 0.5 seconds. The nodes stay in the emergency state for 180 seconds and then go back to the normal state. The same experiment is repeated for 100 times with different node layouts. For the retransmission schedule, the backoff for a critical packet ranges from 0.1 to 0.2 seconds after the first emission and that for an important packet is from 0.25 to 0.3 seconds. In the experiments for a large-scale event, four of the 32 emergency nodes are of the critical class and others are of the important class. Congestion detection is done by not receiving any acknowledgement from any of its next-hop nodes, and the parameters for additive increase and multiplicative decrease in AIMD rate control are set at 0.05 packets/s and 0.5 respectively.

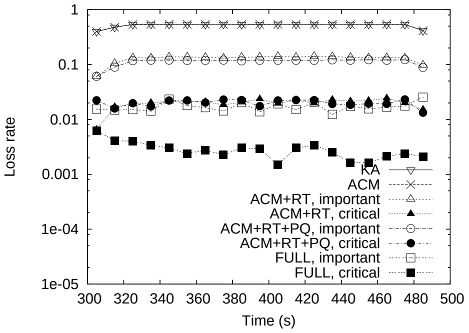

Fig. 8. The loss rate of emergency packets (large-scale event).

B. Results and Discussion

In order to evaluate the effect of mechanisms comprising UMIUSI, we compared five variants of integration of the mechanisms. One is KA (keep awake), in which nodes in a corridor keep awake but neither suppression of normal packets nor other mechanisms is conducted. The second is ACM, in which an assured corridor is established by additionally suppressing emission of normal packets. The third one is ACM+RT (retransmission), in which ACM and the retransmission are applied. The fourth is ACM+RT+PQ (priority queueing), in which the priority queueing is additionally applied. The last one, FULL, employs all of the mechanisms explained in Section III-A.

1) Small-scale event: The loss rate of emergency packets is shown in Fig. 6. The loss rate here is defined as the ratio of emergency packets which did not arrive at a BS out of all emergency packets emitted from a source node. The horizontal axis corresponds to the simulation time. The results of ACM+RT+PQ and FULL are identical to those of ACM+RT, since there is no important class traffic. Therefore, the results of these two variants are not shown in Fig. 6. In the case of KA, keeping intermediate nodes awake without suppression of normal packets, the loss rate rises since collisions with normal packets are unavoidable. On the other hand, in ACM, although the initial rate is the same as that of KA, the loss rate drops gradually in about 30 seconds. This corresponds to time required to establish an assured corridor from a source to the

BS, which further depends on the hop distance of the source node and the sleep schedule of the synchronization-based data gathering scheme. With retransmission, *i.e.*, ACM+RT, there is no packet loss. However, the total number of emergency packets emitted is larger than that of ACM by 15 % and this increase leads to additional energy expenditure.

While the results are not shown because of space limitation, the end-to-end delay with suppression of normal packets in ACM (38.3 ms) is slightly smaller than that in KA (38.5 ms) due to shorter backoff in MAC layer. However, retransmission introduces additional delay in waiting for retransmission and the resultant delay of ACM+RT (41.0 ms) becomes larger than that of KA.

2) Large-scale event: The total throughput of critical and important class averaged over 100 experiments is illustrated in Fig. 7. Here the total throughput is defined as the number of emergency packets received by a BS per second. Note that the plots of the critical class in ACM+RT+PQ and FULL are overlapped to seem like a single plot. With rate control mechanism (FULL), the total throughput of the important class drops gradually in about 40 seconds to be around 5 packets/s, while that without rate control is 50 packets/s. No difference in the total throughput can be seen in the critical class between with and without the rate control. This is because the total traffic of the critical class is 8 packets/s and, as shown later, the loss rate of critical packets is as low as around 2 % even without the rate control. Similarly, little difference was observed between the total throughput of ACM+RT and that of ACM+RT+PQ for both classes, thus the results of ACM+RT are not shown in Fig. 7. After 490 seconds, EMG_SEND nodes go back to the NORMAL state, thus the throughput drops.

The loss rate of emergency packets is shown in Fig. 8. By comparing results of KA and ACM, it can be seen that suppression of emission of normal packets has little effect in a large-scale event as conjectured in Fig. 5, since most of packet loss is caused by collisions among emergency packets. By introducing the retransmission mechanism, the loss rate of critical class traffic is further decreased than important class traffic due to retransmission scheduling. With the rate control mechanisms, *i.e.*, FULL, the loss rate of the critical class is reduced to about 0.25 %, while that of ACM+RT+PQ is around 2 %. Considering this together with the results of Fig. 7, we can say that we sacrifice 90 % of the throughput of the important class in order to make the loss rate of the critical class one-eighth. The AIMD parameters affect this trade off.

The average end-to-end delay is almost the same at 40.8 ms in KA and ACM and they are the smallest among the five variants since there is no retransmission involved. In ACM+RT, the delay rises up in the first 20 seconds to be 147 ms and 198 ms for the critical and important class respectively, reflecting more frequent retransmissions due to collisions among emergency packets. The scheduled retransmission contributes to smaller delay of critical class traffic than that of important class traffic. With the rate control mechanisms, the delay of the critical class gradually decreases to be 67.8 ms, since the number of retransmission and backoff become smaller as the rate control mechanisms come into effect to regulate important class traffic.

These observations in the simulation experiments are consistent with the discussion in Section III-B.

V. Conclusion

In this paper, we presented the methodology for designing a WSN architecture for fast and reliable transmission of urgent information, where several simple and fully-distributed mechanisms working in different spatial and temporal levels are incorporated. We designed a network architecture, called UMIUSI, following the methodology. We verified that UMIUSI successfully improved the delivery ratio and the delay of emergency packets regardless of the scale of emergency.

Acknowledgment

This research was supported by “Special Coordination Funds for Promoting Science and Technology: Yuragi Project” and a Grant-in-Aid for Scientific Research (A)(2) 16200003 of the Ministry of Education, Culture, Sports, Science and Technology in Japan.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, Mar. 2002.

[2] D. Chen and P. K. Varshney, “QoS support in wireless sensor networks: A survey,” in *Proceedings of ICWN 2004*, Las Vegas, Nevada, USA, June 2004, pp. 227–233.

[3] M. Younis, K. Akkaya, M. Eltoweissy, and A. Wadaa, “On handling QoS traffic in wireless sensor networks,” in *Proceedings of HICSS 2004*, Hawaii, USA, Jan. 2004.

[4] Y. Sankarasubramaniam, B. Akan, and I. F. Akyilidiz, “ESRT: Event-to-sink reliable transport in wireless sensor networks,” in *Proceedings of MobiHoc 2003*, Annapolis, Maryland, USA, June 2003, pp. 177–188.

[5] H. Dubois-Ferrière, D. Estrin, and M. Vetterli, “Packet combining in sensor networks,” in *Proceedings of SenSys ’05*, San Diego, California, USA, Nov. 2005, pp. 102–115.

[6] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher, “A spatiotemporal communication protocol for wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 16, no. 10, pp. 995–1006, 2005.

[7] C.-Y. Wan, S. B. Eisenman, and A. T. Campbell, “CODA: congestion detection and avoidance in sensor networks,” in *Proceedings of SenSys ’03*, Los Angeles, California, USA, Nov. 2003, pp. 266–279.

[8] B. Hull, K. Jamieson, and H. Balakrishnan, “Mitigating congestion in wireless sensor networks,” in *Proceedings of SenSys ’04*, Baltimore, Maryland, USA, Nov. 2004, pp. 134–147.

[9] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis, “Interference-aware fair rate control in wireless sensor networks,” in *Proceedings of SIGCOMM ’06*, Pisa, Italy, Sept. 2006, pp. 63–74.

[10] T. Kawai, N. Wakamiya, and M. Murata, “ACM: A transmission mechanism for urgent sensor information,” in *Proceedings of IPCCC 2007*, New Orleans, Louisiana, USA, April 2007, pp. 562–569.

[11] R. Rajagopalan and P. K. Varshney, “Data aggregation techniques in sensor networks: A survey,” *IEEE Communications Surveys and Tutorials*, vol. 8, no. 4, pp. 2–17, 2006.

[12] C.-Y. Wan, A. T. Campbell, and L. Krishnamurthy, “PSFQ: a reliable transport protocol for wireless sensor networks,” in *Proceedings of WSNA 2002*, Atlanta, Georgia, USA, Sept. 2002, pp. 1–11.

[13] A. Woo and D. E. Culler, “A transmission control scheme for media access in sensor networks,” in *Proceedings of MobiCom 2001*, Rome, Italy, July 2001, pp. 221–235.

[14] N. Wakamiya and M. Murata, “Scalable and robust scheme for data gathering in sensor networks,” in *Proceedings of Bio-ADIT 2004*, Lausanne, Switzerland, Jan. 2004, pp. 412–427.