무선 센서-엑터 네트워크에서 주기적 메시지의 실시간 전송

*Dao Quan Ngo, **이상진, **김명균
울산대학교 공과대학 컴퓨터정보통신공학부
*quannd2981@yahoo.com, ** mot60@mail.ulsan.ac.kr, **mkkim@ulsan.ac.kr

Real-Time Communication of Periodic Messages on Wireless Sensor and Actor Networks

*Dao Quan Ngo, **Sang-Jin Lee, **Myung-Kyun Kim
Department of Computer Engineering and Information Technology, University of Ulsan

Abstract

Wireless network technologies are becoming more widely used in industrial environment. The industrial communication system requires a real-time delivery of messages. The sensors periodically senses the physical environment and wants to deliver the data in real-time to the sink. This paper proposes a real-time protocol for periodic messages on wireless sensor and actor networks to be used in industrial communications. The proposed protocol delivers the data message using the shortest path from the source and the nearest actor. The protocol considers the energy consumption by reducing the number of broadcast messages during flooding. We have evaluated the performance of the proposed protocol using QualNet simulator. The simulation results show that the data messages have been delivered in real-time and the number of broadcast messages is reduced from 90% to 35% compared the existing protocols.

1. Introduction

Wireless communication is becoming increasingly important for many industrial automation systems [1]. Sensors and actuators operating over wireless links allow for flexible installation, fully mobile operation, and circumventing the expensive cabling problems. Wireless communication systems for industrial automation, however, require high reliability and real-time data transfer. These requirements are important in industrial automation systems.

WSANs [2][9] consist of a large number of sensors and actors. Sensors sense the physical environment such as temperatures, humidity, motions, vibrations, etc., and transmit the data to the actor or sink. Sensors are usually static devices with limited capacities of power, bandwidth and computation. On the other hand, actors move throughout the sensor field and are resource-rich devices with higher processing and transmission capabilities as well as longer battery life.

Some recent papers [3][5][6] have considered the issue of real-time communication in sensor networks. The RAP [3] provides service differentiation in the timeliness domain by velocity-monotonic classification of packets. SPEED protocol [5] provides real-time communication services and is designed to be a stateless, localized algorithm with low control overhead. MMSPEED [6] is an extension of SPEED that can differentiate between flows with different delay and reliability requirements. Those approaches use the geographical information to determine the packet scheduling or the packet transmission speed. But, installing the location system to each sensor is expensive in a densely deployed large wireless sensor network.

The real-time protocol for periodic messages proposed in this paper is source-initiated and consists of a REQ-REP-DATA transaction. When a source wants to send a new periodic message, it broadcasts a REQ (REQuest) message with TTL (Time To Live) value. The REQ messages are flooded within TTL hop counts. The proposed protocol uses a RSSI-based flooding to find the shortest path from the source and the actor and to reduce the number of broadcasted REQ messages. When the actor receives the REQ message, it responses by REP (REPly) message, and it is forwarded toward the source. During the exchange of REQ and REP messages, the shortest path from the source and the actor is established. If the source receives the REP message, it transmits the DATA message periodically to the actor. The proposed protocol delivers periodic DATA messages in real-time using the shortest path from the source and the actor and minimizes the energy consumption by reducing the number of broadcasted REQ messages. Our simulation experiments demonstrate that, for WSAN, the proposed protocol delivers the DATA messages using the shortest path from the source and to the actor. It also reduces the energy consumption by reducing the percentage of nodes that relay messages from 90% to 35%, compared to existing flooding protocols.

The rest of this paper is organized as follows. The related work is described in Section 2, and Section 3 describes the architecture and the components of wireless sensor and actor networks. The real-time protocol for periodic messages is described in Section 4. Simulation results are presented in Section 5 and Section 6 concludes the paper with a summary.

2. Related Works

Several real-time protocols have been proposed for sensor networks. RAP [3] provides service differentiation in the timeliness domain by velocity-monotonic classification of packets. Based on packet's deadline and destination, its required velocity is calculated and its priority is determined in the velocity-monotonic order so that a high velocity packet can be delivered earlier than a low velocity one. Implicit earliest deadline first (I-EDF) [4] can provide hard real-time guarantee based on decentralized EDF packet scheduling. However, it works only when most traffic is periodic and all periods are known a priori, which is not the case for many sensor network applications.

The Stateless Protocol for Real-time Communication in sensor networks (SPEED) [5] provides real-time communication services and is designed to be a stateless, localized algorithm with low control overhead. End-to-end soft real-time communication is achieved by maintaining a desired delivery speed across the sensor network through a combination of feedback control and non-deterministic geographic forwarding. Multi-path and Multi-Speed Routing Protocol (MMSPEED) [6] for probabilistic QoS guarantee in wireless sensor networks. Multiple QoS levels are provided in the timeliness domain by guaranteeing multiple packet delivery speed options, while various requirements are supported by probabilistic multi-path forwarding in the reliability domain. SPEED and MMSPEED try to provide real-time delivery of individual flows from different sensors. Both of these protocols use geographical information to determine the delay to the neighbors and the speed of the message to transmit, which is expensive, but the location system is expensive to install for cheap sensors.

Li and Shenoy [7] propose scheduling communication solution that explicitly avoids network collisions and minimizes the completion time to transmit a set of sensor messages. It presents three heuristics, based on edge coloring, to achieve QoS requirements. Scheduling messages with deadlines solution [8] drives the effective deadline and the latest start time for per-hop message transmissions from the validity intervals of the sensor data and the constraints imposed by the consuming task at the destination. It exploits spatial channel reuse for each per-hop transmission to avoid collisions. But [7] and [8] are both centralized solutions, so that these are not suitable to apply for large-scale networks.

Existing researches are not suitable for high density network with frequent situation which can cause collision and interference when too many sensors report and relay data to destinations. Our proposed protocol tries to eliminate the unnecessary transmission in the networks for supporting soft real-time communication.

3. Wireless Sensor and Actor Networks (WSANs)

The wireless sensor and actor networks to monitor the work environment in factory consist of a large number of sensors and actors as shown in Fig. 1. Each sensor senses the physical environment and transmits its data to the actors. When an actor receives the data from the sensor, it performs some actions or transmits the data to the sink for the sensor network applications. In our sensor networks, sensors are stationary and densely deployed to cover the entire sensing area. Actors are mobile and the data sensed by the sensors are collected through the nearest actors. The performance of the network depends on the number of actors in the network. It has been shown that the optimum number of actors is 4 [10].

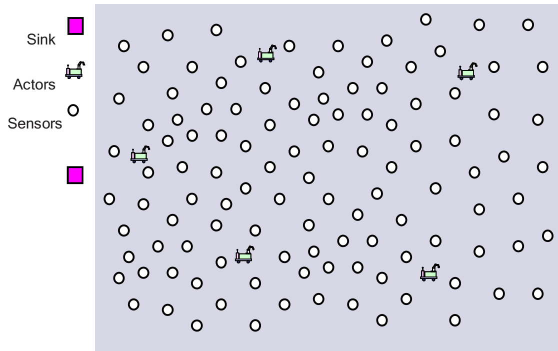

Fig. 1. A wireless sensor and actor network.

When a sensor detects an event from the environment, it senses and transmits the event data periodically for some time and the data needs to be delivered to the nearest actor in real-time. We should also consider energy consumption in our protocol because the energy is the scarcest resource in wireless sensor networks.

4. Wireless Sensor and Actor Networks (WSANs)

For real-time message delivery, the proposed protocol finds a route from the sensor to the nearest actor on demand. The proposed protocol in this paper is a source-initiated protocol where the source starts the route discovery when is has a new periodic message to transmit. The protocol consists of the following steps.

- When a source needs to send a new periodic message, it floods a REQ (REQuest) message through the network. The REQ message is broadcasted over the network and one or more actors will receive the message.
- If an actor receives the REQ message, it responds by a REP (REPly) message in response to the REQ message. The routing path is established while exchanging the REQ and REP messages.
- When the source receives the REP message, it transmits DATA messages to the actor through the path established during the REQ and REP messages exchange.

4.1 Flooding of REQ messages

When a sensor wants to transmit a new periodic message, it broadcasts over the network a REQ message to find a route from itself to the nearest actor. The REQ message includes a specification for the periodic message such as the source ID (*srcID*), the message ID (*msgID*), the message type (*msgType*), the period of the message (*msgPd*), and a hop count (Time To Live) (*TTL*). The source and message IDs identifies a periodic message uniquely and each periodic message has its own deadline and period. The TTL value specifies a hop count and is used to limit the area within which the REQ message is broadcasted. The TTL value is initialized to a proper value by the source, and when a sensor receives the REQ message, it decrements the TTL value by one and tries to broadcasts again the message if the TTL value is not zero and drops the message if it becomes zero.

When a sensor broadcasts the REQ message, all of the nodes within its transmission range will receive the message. If all of the nodes that received the REQ message try to broadcast the message again, there will be a large number of REQ broadcast messages and collision probability will become high. So, we need to reduce the number of broadcast messages. Besides, to deliver the data message in real-time, the hop count of the routing path from the source to the actor needs to be small.

To control the number of REQ broadcast messages and to find a path to deliver data messages from the source to the actor in real-time, the proposed protocol uses a RSSI (Received Signal Strength Indicator)-based flooding. In the RSSI-based flooding, if a sensor receives a REQ message, it checks whether the REQ message has already been received before or not. If the message is a new one, then it decrements the TTL value and if the TTL is still greater than 0, it tries to broadcast the REQ message again after waiting for certain amount of time (TO: TimeOut). While waiting for TO time, if it hears other node broadcast the REQ message, it cancels broadcasting the REQ message. The TO value is calculated as in (1) to be inversely proportional to the RSSI of the REQ message.

$$\mathrm{TO} = \left(\frac{\mathrm{r} - \mathrm{R}_{\min}}{\mathrm{R}_{\max} - \mathrm{R}_{\min}}\right) \cdot T \qquad (1)$$

In (1), R_{min} is a minimum level of the received signal strength above which a sensor can decode the message reliably. R_{max} is the transmission power strength and r is the received signal strength. T is a maximum timeout value which is set accordingly depending on the application.

Node x maintains MQ (Message Queue) which contains a set of received REQ messages. Each entry of MQ consists of (REQ_M, TO, P_{rev}, P_{fwd}, T_{rev}, T_{fwd}). TO is a timer for re-broadcasting the received REQ message which is calculated as in (1). P_{rev} represents the node from which node x received the REQ_M message, and P_{fwd} represents the next node to which node x can transmits DATA message toward the actor. T_{rev} represents a timer after which the reverse path P_{rev} is timed out and T_{fwd} represents a timer after which the forward path P_{fwd} is timed out.

If TO of REQ_M message expires, node x sets up a reverse path P_{rev} as the node from which it received the REQ_M message and transmits the REQ_M broadcast message.

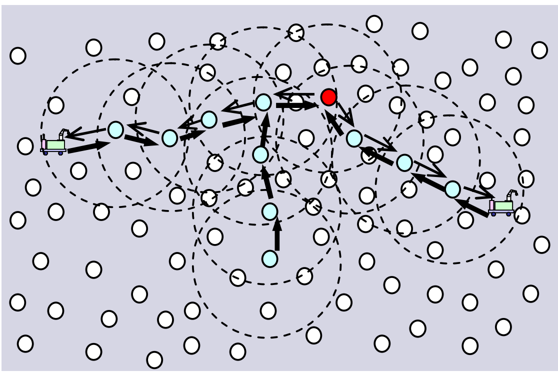

Fig. 2. An example of REQ-REP-DATA message transaction.

If an actor receives the REQ message, it transmits a REP (REPly) message to the node from which it received the REQ message and stays there until it finishes the transaction (REQ-REP-DATA transmissions). The REP message is delivered to the source according to the reverse path set up during flooding the REQ message. The forward path is set up in a similar way as the reverse path while the REP message is delivered to the source. When the source receives the REP message, it starts to transmit DATA messages periodically according to the forward path set up during the REP message delivery. The source can receive the REP message from one or more actors if there are multiple actors within TTL hop counts. If the source receives more than one REP messages, it chooses the first REP message. Fig. 2 shows an example of REQ-REP-DATA message transmissions where the TTL value was set to 5 by the source.

4.2 Timer maintenance

Each sensor node maintains MQ which has a set of received REQ messages. Each entry contains a REQ_M message, the information about the forward and reverse paths and the timers for the paths. Timers are used to reduce the overhead of table maintenance by invalidating the unnecessary entries.

Reverse paths are used to deliver the REP message toward the source, so the timer for the reverse path, T_{rev}, needs to be maintained only until the node receives the REP message. When a sensor node creates a new MQ entry for the received REQ_M message, it initializes T_{rev} to Max_E2E_Delay which is a given maximum end-to-end delay in the network. After creating the entry, if the node does not receive a REP message until T_{rev} expires, the MQ entry is invalidated.

Forward paths are used to deliver the DATA message toward the actor, so the timer for the forward path, T_{fwd}, needs to be maintained while the source is transmitting the DATA messages to the actor. The DATA message is transmitted from the source to the actor periodically in *msgPd* time interval. When a sensor node receives a REP message for the REQ message transmitted before, it initializes the forward timer T_{fwd} to (*msgPd* + Δ) and T_{fwd} is re-initialized to (*msgPd* + Δ) each time it receives a DATA message toward the actor. Even though the source transmits the DATA message regularly in *msgPd* time interval, there will be a little delay variance due to the message traffic in the network. Δ is used to consider the message delay in the network and is set properly according to the amount of traffic. The proposed real-time message transmission protocol using RSSI-based flooding is described in the following.

Real-time protocol using RSSI-Based Flooding

// Request message, *REQ_M*, contains (*srcID*, *msgID*,
// *msgType*, *msgPd*, *TTL*)

when (node *x* receives a REQ message from node *y*)
{
  TTL = *TTL* - 1;
  if (<*srcID, msgID*> is in MQ **or** *TTL* == 0) {

```
        Drop the recieved REQ message; exit;
    };
    Choose an empty slot in MQ and store REQ_M;
    TO = T * (r - Rmin)/( Rmax - Rmin);
    Prev = y;   Pfwd = NULL;
    Trev = Tfwd = 0;
}

when (node x receives a REP message from node z) {
    Find REQ_M entry for the REP message in MQ;
    Pfwd = z;
    Tfwd = msgPd + Δ;
    Transmit the REP message to Prev;
}

when (node x receives a DATA message from node y) {
    Find REQ_M entry for the DATA message in MQ;
    Tfwd = msgPd + Δ;
    Transmit the DATA message to Pfwd;
}

when (TO expires for REQ_M message in MQ) {
    Trev = Max_E2E_Delay;
    Re-broadcasts REQ_M message;
}
------------------------------------------
```

The proposed protocol has the following features.

- It can deliver DATA messages in real-time because it uses the shortest path from the source to the nearest actor.
- It minimizes the energy consumption by reducing the number of messages during REQ message flooding using RSSI.
- It is simple and the route information is found when needed and maintained only during the transaction.

5. Performance Evaluation

The performance of the proposed protocol has been evaluated by comparing with normal flooding protocol through simulation using QualNet 3.9 [2]. We use notation NFP to indicate normal flooding protocol, and RBFP to indicate our real-time protocol using RSSI-based flooding protocol. Sources are randomly chosen and destinations are randomly placed into network. Each experiment was performed seven times and then average value for each metric is presented. Table 1 describes the detailed setup for our simulator.

Table 1: Simulation Conditions

Simulation time	10s
Dimension	100 x 100 m2
Transmission range	15 m
Packet size	128

RBFP is evaluated with variation of number of nodes as 50, 75, 100, 125, 150, 175 and 200. All nodes are uniform placed in an area. Fig. 3 shows RSSI variation according to distance between sender and receiver which are set up in the network.

In Fig. 4, RBFP shows a good percentage of relay nodes which is improved from 50% to 75% in comparison with NFP when number of nodes varying from 50 to 200. When network density increases, then percentage of relay nodes reduces because there are more nodes in transmission range of sender so that number of nodes receiving message from farther one increases. RBFP is, of course, better than NFP in this metric because closer nodes have low probability to relay messages. From the result, we can imagine improvement of energy consumption which is proportional to percentage of relay nodes.

Fig. 5 shows that NFP and RBFP have almost the same delivery ratio. NFP is little higher than RBFP as network density is low, because it is not enough nodes to relay message. But RBFP is higher than NFP when network density is high because collision happens in NFP.

In Fig. 6, RBFP is showed that it has good end-to-end delay over the other. In NFP, the number of hops in route path increases according to network density, so that end-to-end delay is proportional to network density. But RBFP is different from NFP; route path in RBFP can be short when increasing network density, because the distance of one hop can increase as there are many nodes in border of node transmission range. Conversely, when network density is low, nodes that are close to sender can become members of route path, so the distance of a hop can be small. Hence, RBFP end-to-end delay is inversely proportional to network density.

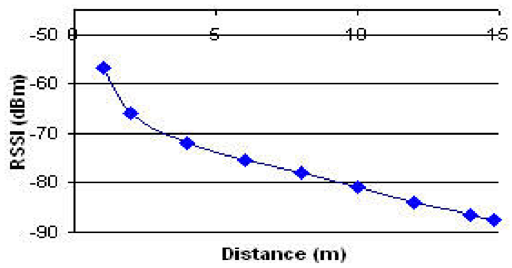

Fig. 3. RSSI Variation

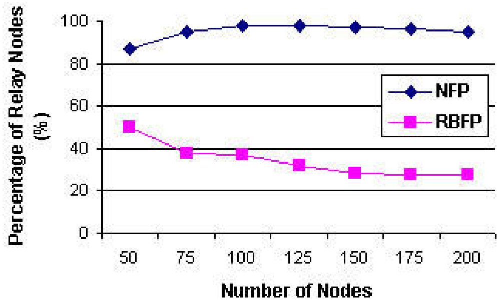

Fig. 4. Percentage of Relay Nodes

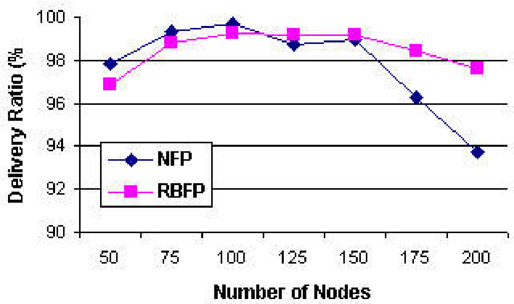

Fig. 5. Delivery Ratio

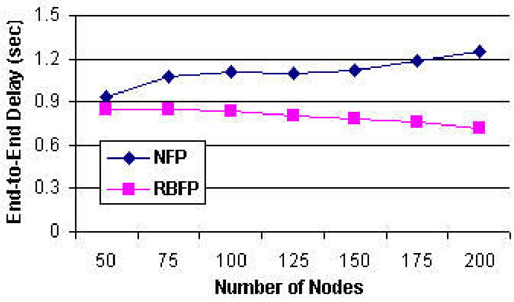

Fig. 6. End-to-End Delay

6. Conclusions and Future Works

Wireless network technologies are becoming more and more widely used in industrial automation systems. However, the industrial communication systems require real-time and reliable data delivery. The sensors periodically senses the physical environment and needs to deliver the data to the sink in real-time. This paper has proposed a real-time protocol for periodic messages on wireless sensor and actor networks, which can be used in industrial environment. The proposed protocol has the following features.

- It delivers the DATA messages in real-time using the shortest path from the source and the nearest actor.
- It considers the energy consumption by reducing the number of broadcast messages during flooding.
- It is simple and reduces the routing path maintenance overhead by keeping the path information only when needed.

References

[1] A. Willig, K. Matheus, and A. Wolisz, "Wireless technologies in industrial networks," Proceedings of the IEEE, vol. 93, no. 6, pp. 1130-1150, June 2005.

[2] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet", In Proc. of ASPLOS-X, October 2002.

[3] C. Lu, B.M. Blum, T.F. Abdelzaher, J.A. Stankovic, and T. He, "RAP: A Real-Time Communication Architecture for Large-Scale Wireless Sensor Networks", Proc. IEEE Real-Time and Embedded Technology and Applications Symp. (RTAS 2002), Sept. 2002.

[4] M. Caccamo, L. Zhang, L. Sha, and G. Buttazzo, "An Implicit Prioritized Access Protocol for Wireless Sensor Networks", Proc. IEEE Real-Time Systems Symp. (RTSS '02), pp. 39-48, 2002.

[5] T. He, J. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks", Proc. IEEE Int'l Conf. Distributed Computing Systems, pp. 46-55, 2003.

[6] E. Felemban, C.-G. Lee, E. Ekici, "MMSPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks", IEEE Transactions on Mobile Computing, vol. 5, no. 6, June 2006.

[7] H. Li, P. Shenoy, K. Ramamritham, "Scheduling Communication in Real-Time Sensor Applications", IEEE Real Time Technology and Applications Symposium (RTAS), May 2004.

[8] H. Li, P.Shenoy, K. Ramamritham, "Scheduling Messages with Deadlines in Multi-hop Real-time Sensor Networks", IEEE Real-Time and Embedded Technology and Applications Symposium, San Francisco, California, March 2005.

[9] Gungor, V. C., Akan, O. B., and Akyildiz, I. F., "A Real-Time and Reliable Transport Protocol for Wireless Sensor and Actor Networks," to appear in IEEE/ACM Transactions on Networking for publication, June 2008.

[10] Gungor, V. C., Vuran, M. C., and Akan, O. B., "On the Cross-Layer Interactions between Congestion and Contention in Wireless Sensor and Actor Networks," to appear in Ad Hoc Networks Journal (Elsevier), August 2007.

[11] http://www.scalable-networks.com/