Sensor-Actuator Communication Protocols in Wireless Networks

Kiyohiro Morita[1], Keiji Ozaki[1], Kenichi Watanabe[1], Naohiro Hayashibara[1], Tomoya Enokido[2], and Makoto Takizawa[1]

[1]Tokyo Denki University, Japan
{kiyo,kei,nabe,haya}@takilab.k.dendai.ac.jp,
makoto.takizawa@computer.org
[2]Rissho University, Japan
eno@ris.ac.jp

Abstract. A wireless sensor-actuator network (WSAN) is composed of sensor and actuator nodes interconnected in a wireless channel. Sensor nodes can deliver messages to only nearer nodes due to weak radio and messages are forwarded by sensor nodes to an actuator node. Messages sent by nodes might be lost due to collision and noise. We discuss the redundant data transmission (RT) protocol to reduce the loss of sensed values sent to an actuator node even if messages are lost. In the RT protocol, a sensor node sends a message with not only its sensed value but also sensed values received from other sensor nodes. Even if a message with a sensed value v from a sensor node is lost, an actuator node can take the value v from other messages. We evaluate the RT protocol compared with the CSMA protocol in terms of how much sensed values an actuator node can receive in presence of message loss.

1 Introduction

A wireless sensor-actuator network (WSAN) [1,2,7] is composed of sensor nodes and actuator nodes interconnected in a wireless network. A sensor node sends sensed values to actuator nodes. An actuator node performs methods on actuation devices. If multiple nodes simultaneously send messages, the messages are lost due to collision. Synchronization mechanisms [3,8,9] are discussed to reduce message loss. However, only simple mechanisms like CSMA [3] are used in sensor nodes. In addition, the farther a destination node is from a source node, the more number of messages are lost due to noise in a wireless channel. Since sensor nodes can send only week radio, the more number of messages are lost due to noise the IEEE 802.11 [10].

In this paper, we discuss the *redundant data transmission* (RT) protocol to reduce the loss of sensed values where sensed values and control information in a message from a sensor node are redundantly transmitted in other messages. In the RT protocol, sensed values can be delivered to an actuator node even if a message with the sensed values is lost. We evaluate the RT protocol in terms of the number of sensed values delivered to an actuator node in presence of message

T. Enokido, L. Barolli, and M. Takizawa (Eds.): NBiS 2007, LNCS 4658, pp. 11–19, 2007.

loss caused in the traditional CSMA [3] based data transmission. The loss ratio of sensed values can be reduced to about 70% even if messages are lost.

In section 2, we discuss the RT protocol. In section 3, we evaluate the RT protocol in terms of data loss ratio.

2 Redundant Data Transmission (RT) Protocol

2.1 Message Format

We consider a wireless sensor-actuator network (WSAN) [4] which is composed of multiple *sensor* nodes $s_1, \cdots, s_n$ and one *actuator* node a which are interconnected in wireless channel. A sensor node gathers information in the physical world and then sends the sensed values to actuator nodes. Then, an actuator node a performs action from the sensed values on actuation devices. Typically, a sensor node can deliver messages to nodes in at longest 5 $[m]$ while an actuator node can deliver to nodes in 100 $[m]$ [11]. Every sensor node may not directly deliver a message to an actuator node. A sensor node forwards a message to other nodes in multi-hop communication. Each node transmits messages with the CSMA [3] transmission-scheme. Here, a node first listens to the channel. If it is idle, the node starts transmitting a message. Otherwise, the node waits for some time units.

Let us consider three nodes s_1, s_2, and s_3. Suppose the node s_1 sends a message m_1 and s_2 receives m_1. Suppose, s_3 does not receive m_1 due to noise and collision as shown in Figure 1. Suppose s_2 sends a message m_2 after receiving m_1 and s_3 receives m_2. Here, if the message m_2 carries data d_1 in m_1 to s_3, s_3 receives not only m_2 but also the data d_1 of the lost message m_1. Data of another message carried by a message is referred to as *backup data*. A sensor node s_i obtains a sensed value v and then sends a message m of the following format:

[Sensor message m**]**

- $m.src$ = source sensor node s_i of the message m.
- $m.seq$ = sequence number of the message m.
- $m.val$ = value v sensed by the sensor node s_i.
- $m.state = ON$ if the sensor node s_i knows that $m.val$ is received by an actuator node a, else OFF.
- $m.data$ = *backup* data $\langle data_1, \cdots, data_K \rangle$.
- $m.data_j$ = *backup* tuple $\langle sid, seq, val, state \rangle$ $(j = 1, \ \cdots, \ K)$.

If a sensor node s_i sends a message m after sending another m_1, $m.seq = m_1.seq + 1$. In addition to sending the sensed value $m.val$, m carries *backup* data in $m.data$ which are sensed values which s_i has received from other sensor nodes. $m.data$ include the number K of *backup* tuples. For each *backup* tuple $d = \langle sid, seq, val, state \rangle$ in $m.data$, $d.state = ON$ if s_i receives the confirmation of the sensed value $d.val$, which is sent by $s_j (= d.sid)$ from an actuator node a. Here, the message m sent by s_i is referred to as *backup* message of m_j sent by another sensor node $s_j (= d.src)$ if $m_j.seq = d.seq$ and $d \in m.data$. If a node