A Publish-Subscribe Middleware for Real-Time Wireless Sensor Networks

Mohsen Sharifi, Majid Alkaee Taleghan, and Amirhosein Taherkordi

Computer Engineering Department, Iran University of Science and Technology
msharifi@iust.ac.ir, {alkaee, taherkordi}@comp.iust.ac.ir

Abstract. The specific characteristics of Wireless Sensor Networks (WSNs) have changed Quality of Service (QoS) support in these networks to a challenging task. In this paper, we propose a dispatcher as part of a message routing component in publish/subscribe WSNs. Some works consider link-based solutions to support real-time parameters in WSNs. These works do not take into account the dynamic behavior of WSNs with probable damaged nodes and links. The use of dispatcher can reduce the average message delay, whether the message has high priority or low priority. The dispatcher uses a scheduler to support real-time parameters, such as delay, and selects messages from two separate queues, namely, QoS queue and non-QoS queue. Simulation results show that our approach really reduces the average delay and increases the delivery rate for both QoS messages and non-QoS messages.

1 Introduction

Due to real-time requirements, high degree of faults, noise, and non-determinism caused by the uncontrolled aspects of environment [8], a fundamental issue in realizing a WSN is how to route applicative information, i.e., messages controlling the operation of various sensors and the data gathered by them. In WSNs' context, few works consider QoS parameters and especially real-time parameters, such as, delay in their solutions [2].

Most of works [3, 6] consider the sink node as an interface between a WSN and user applications, so that there is only one sink node in the WSN, which has higher capabilities and does not have the constraints that the other nodes have. They assume the links between nodes do not change frequently and the networks have a specific organization. A recent work [2] considers the sink as a *role* that can be assigned to every one of nodes, and not necessarily only the node which has higher resources.

Some protocols with QoS support at network layer, require much energy, are too complex, and consider only one QoS factor, i.e. either fault tolerance or real-time [1]. A more recent solution [4] considers both of these two QoS factors, but it does not support the data-centric style of communication and energy-awareness.

Mires [7] is a message-oriented middleware and needs a pre-configuration phase to detect publishers and so it cannot operate in dynamic environments with changing behavior. It does not consider QoS parameters in its design and implementation too.

The semi-probabilistic approach [2] has proposed an approach to route events based on a semi-probabilistic broadcast schema. It shows a semi probabilistic way against flooding approach to transmit the WSNs' messages. It is one of the

(probabilistic) approaches proposed in [5] for mobile ad-hoc networks. The semi-probabilistic approach has not taken into account real-time properties in its work and uses probability when a node cannot find any subscription for sending event. Therefore, this approach has its own overheads too.

In this paper, we propose an approach to consider the delay time from publishing time (start time) until when they are received by subscribers (end time). The approach uses the concepts from the semi-probabilistic approach. It does not need the configuration phase of nodes that some approaches like cluster-based approaches require. QoS parameters are injected in interests, and subscribers send their interests to network. Our approach uses *QoS-queue* and *non-QoS-queue* to buffer QoS and non-QoS messages, respectively, in order to reduce the delay times of messages and to manage these messages as quickly as possible. This means that QoS messages are scheduled according to their real-time constraints.

The remainder of paper is organized as follows. Section 2 describes our approach. Section 3 is dedicated to simulative comparison of our approach with the work reported in semi-probabilistic approach. Section 4 concludes the paper.

2 Our Approach

According to what has been proposed in the semi-probabilistic approach, frequent changes in WSNs do not allow considering the links concept between nodes. In publish-subscribe based systems, both subscribers and publishers can be QoS-aware and use QoS parameters to distinguish QoS messages. Subscribers can add a QoS parameter (e.g. delay) to interests, and publishers can add priority to detected events when they detect or sense an object.

There are two steps in our mechanism; the first step is the dissemination of interests, and the second step is the dispatching of events. After an interest is specified, one copy of the interest is stored in *local-subscription* storage and the interest is published to the network. Neighboring nodes, which are in subscriber's radio range, receive the interest. When a node receives the interest, it stores the interest in its *non-local-subscription* storage.

We use a value called *subscription horizon* which denotes the range that the interests are broadcasted and is the hop count; the same value is considered in the semi-probabilistic approach. In the event propagation step, publishers publish events according to their available sensors and operation specification of events. Every node has a QoS messages queue and a non-QoS messages queue.

The remaining part of our mechanism is the dispatching of messages based on their priorities and delays. These two parameters help the scheduler to send messages. Events are removed from queues based on the priority of their subjects. They are checked against local-subscription and non-local-subscription. If the event matches the local-subscription, a subscription node receives it. If the event matches to the non-local-subscription, it is re-broadcasted to other nodes. Otherwise, it uses *event propagation threshold* and if it matches, it re-broadcasts the event.

Since the re-broadcast of message can produce duplicate messages in the network, we use a message identifier which consists of publisher identifier and the time the event has been published; the publisher identifier is a local identifier in vicinity. When a message is forwarded and removed from a node's queues, it may be possible that it

receives the same message again. Thus, we keep the publisher identifier and the last event time for that received publisher event to prevent re-broadcasting of older messages that are looped back to this node.

3 Simulation and Evaluation

To evaluate how well our mechanism supports real-time QoS in WSNs, the JiST simulator (<http://jist.ece.cornell.edu/>) was used. The simulation filed was 120 meters x 120 meters and consisted of 40 nodes with radio range 40 meters, and simulation was run for 40 seconds. We considered two criticality levels based on subscribers' interests: *High* and *Low*, and used 2 subscribers and 4 publishers. The publisher average rate and the average message transmission time were nearly 2 events per second and 1 second, respectively. The propagation threshold and subscription horizon in the two approaches were 0.5 and 2, respectively.

The semi-probabilistic approach does not consider QoS properties when messages are forwarded. But, in our approach, we consider QoS properties and the QoS scheduler uses these properties to dispatch messages. The scheduler operates according to *first delay* selection criteria. The message delay is measured by deducting the arrival time from the generation time of the message. Fig. 1 shows the delivery and the average delays of messages in both approaches.

The results show that our approach really reduces the average delay (i.e. the time between an event is generated until it is received by a subscriber) and increases the delivery rate (i.e., the ratio between the expected and actual receipt time of events) for both QoS messages and non-QoS messages.

Message waiting time in queue can change the message overheads and consequently the delay. In broadcast-based communication, one message may be in network for a long time and increase the overheads thereof. We have used a message identifier to remove message duplication in queues.

Fig. 1. The delivery rate and average delay of QoS messages and non-QoS messages

4 Conclusion and Future Works

QoS consideration in large-scale distributed middleware systems can raise the applicability of these systems in a variety of domains with favorable results. Unfortunately,

few works have been reported which consider real-time deadlines in the context of WSNs. We considered this quality parameter and the goal was to reduce delays in message transmissions between nodes of WSNs.

We used two queues and a method to add messages to QoS queue and non-QoS queue. We evaluated our scheduler with first delay deadline and showed the average delay and delivery. Evaluation results showed lower average delay and higher delivery rates for QoS messages and non-QoS messages.

We intend to carry out more simulations in future and include more QoS parameters into published messages in order to reduce the duplicate message propagations in the network.

References

- [1] Chen, D., Varshney, P.K.: QoS support in wireless sensor networks: a survey. In: Proceedings of the International Conference on Wireless Networks (ICWN), Las Vegas, USA, (2004).
- [2] Costa, P., Picco, G.P., Rossetto, S.: Publish-subscribe on sensor networks: a semi-probabilistic approach. In: Proceedings of the 2nd IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS), Washington DC, USA, (2005).
- [3] Delicato, F.C., et al.: A service approach for architecting application independent wireless sensor networks. In: Cluster Computing, Springer Science, (2005), 211-221.
- [4] Felemban, E., et al.: Probabilistic QoS guarantee in reliability and timeliness domains in wireless sensor networks. In: Proceedings of IEEE INFOCOM, vol.4, (2005), 2646-2657.
- [5] Ni, S., et al.: The broadcast storm problem in mobile ad hoc networks. In: Proceedings of ACM Mobicom, (1999).
- [6] Sharifi, M., Taleghan, M.A., Taherkordi, A.: A middleware layer mechanism for QoS support in wireless sensor networks. In: 5th International Conference on Networking (ICN), Mauritius, (2006).
- [7] Souto, E., et al.: A message-oriented middleware for sensor networks. In: Proceedings of the 2nd Workshop on Middleware for Pervasive and Ad-Hoc Computing, Canada, (2004).
- [8] Stankovic, J.A., et al.: Real-time communication and coordination in embedded sensor networks. In: Proceedings of the IEEE, vol. 91, no. 7, (2003).