AMoQoSA: Adaptive Modular QoS Architecture for Wireless Sensor Networks

Evy Troubleyn Eli De Poorter Ingrid Moerman Piet Demeester
Ghent University - IBBT, Department of Information Technology (INTEC)
Gaston Crommenlaan 8, bus 201, 9050 Ghent, Belgium
Email: evy.troubleyn@intec.ugent.be

Abstract

In contrast to traditional wireless sensor networks, where energy conservation is the main objective, next-generation wireless sensor networks will additionally demand guarantees in terms of delay, reliability and bandwidth. However, current sensor network protocols offer very limited QoS support. Moreover, the current research does not take into account the growing heterogeneous nature of sensor nodes and applications. Therefore, we present in this paper an Adaptive Modular QoS Architecture (AMoQoSA), which uses several QoS techniques that can be activated according to the nodes' capabilities. This enables the network to dynamically adapt itself to the current network situation in order to continuously deliver optimal QoS guarantees in heterogeneous sensor networks.

1 Introduction

The research in wireless sensor networks has known an unprecedented growth. For many years, wireless sensor networks have been used for simple monitoring purposes, for controlling industrial processes and for automation purposes. This research has been dominated by energy considerations, while Quality of Service (QoS) parameters such as delay and bandwidth were often deemed unimportant and consequently disregarded. Today, wireless sensor network applications demand more and more provisions from the underlying sensor network. Current sensor networks are even used to transport audio and video streams [9]. As a result, QoS requirements on delay, jitter, bandwidth and reliability can no longer be ignored.

Recently, several MAC and routing protocols have been proposed which support QoS [1]. They focus mainly on a layered network approach and QoS constraints such as timeliness [2, 6] and reliability [5]. However, a one-size-fits-all solution is not available: the protocols support only a few of the application requirements and no mention is made of other requirements such as jitter and required bandwidth.

Moreover, the heterogeneity of the sensor network is generally ignored by applying identical QoS mechanisms on each sensor node. Finally, they cannot deal with diverse traffic streams [3] and, consequently, the heterogeneity of future wireless sensor network applications is also ignored.

In [8], a cross-layer framework for optimizing global power consumption and balancing the load in sensor networks through greedy local decisions is presented. This framework is able to adapt the routing and MAC layer behavior of the sensor nodes and, although this framework is originally intended for energy conservations, an extension towards QoS can be made. However, only local adjustments are made in order to improve global network performance. Additionally, this framework applies identical techniques on each sensor node, which is not effective when coping with the heterogeneous nature of sensor networks.

In this paper, we present an Adaptive Modular QoS Architecture (AMoQoSA) that can handle the heterogeneous nature of future sensor networks and applications. This framework enables the network to adaptively change its QoS behavior in order to continuously deliver the most optimal QoS guarantees with respect to energy considerations.

The remainder of this paper is organized as follows. In section 2, we start by discussing how dynamic and heterogeneous sensor networks can be supported at an architectural level. Next, in section 3, we describe three QoS techniques that are scalable towards dynamic and heterogeneous networks. In section 4, an Adaptive Modular QoS Architecture (AMoQoSA) is presented, in which the previously presented QoS techniques can be integrated. Finally, we conclude the paper by giving a short summary of our work in section 5.

2 Supporting dynamic and heterogeneous sensor networks

Sensor networks have been observed to become more and more heterogeneous [3, 11]. This heterogeneity expresses itself in 2 different ways: (i) the capabilities of the

sensor nodes can be different and (ii) the network can support multiple services with different requirements.

2.1 Different node capabilities

Nodes in a sensor network can differ in regard to *energy capacity* (battery powered versus line-powered), *computational power* and *communication capabilities*. Applying the same techniques and mechanisms on each sensor node is not very efficient and against the heterogeneous nature of sensor networks. Therefore, it would be beneficial to apply different techniques on different sensor nodes, based on their remaining energy and their computation and communication capabilities. This way, nodes with advanced capabilities can fulfill more demanding tasks such as data-aggregation, while others only have to fulfill some basic tasks, resulting in a better spread of the energy-usage over the network.

To support these different types of nodes, several QoS profiles can be used to organize the nodes into classes with similar capabilities. These profiles specify which QoS techniques should be used in each type of node. As an example, we give a classification in 3 classes, each supporting QoS at a different level.

2.1.1 Lightweight QoS profile. The lightweight QoS profile is very simple. Sensor nodes implementing this profile do not actively exchange information with neighboring nodes, but can receive information and instructions from more advanced nodes. QoS support is limited to the altering of parameters of passing packets, for example by changing the packet's priority.

2.1.2 Basic QoS profile. A basic QoS-enabled node is capable of exchanging information with neighboring nodes. It has thus a more local network view. Additionally, to support QoS, these nodes can adjust the parameters of the network protocols.

2.1.3 Advanced QoS profile. Finally, the nodes implementing an advanced profile are decision makers: they can communicate with nodes through the whole network to obtain network-wide information. In addition to the above QoS decisions, they can decide to dynamically replace network protocols or enable data-aggregation, thus ensuring global QoS requirements are fulfilled.

2.2 Dynamic requirements

In contrast to traditional wireless sensor networks which are typically used for one application, evolved sensor networks must be able to support many diverse applications on the same network. This means that the network can not be optimized in terms of one application. Instead, the number of applications and the requirements of the applications

will change during the lifetime of the network. Thus, the optimal network configuration (type of protocols, protocol parameters, QoS requirements, ...) will also change. To support these dynamic and changing applications, the supporting QoS protocols should be able to dynamically adapt the behavior of the network.

3 Adaptive QoS techniques

As has been stated in section 2, efficient QoS support for different sensor network applications in a heterogeneous sensor network environment cannot be achieved by simply applying the same techniques on each sensor node. Thus, QoS-enabling techniques should be very adaptive. To support this adaptivity, we make a separation between QoS techniques according to their complexity¹. As such, QoS techniques can be added according to the nodes' capabilities. Three such QoS techniques will be described in the following sections.

3.1 Per-packet inspection

In traditional wireless sensor networks, data packets entering a sensor node are directly sent to the routing protocol. Although this approach is sufficient for simple wireless applications with limited QoS constraints, it is more beneficial to actively process and adapt incoming packets and their properties. When using '*Per-Packet Inspection*', every incoming data packet is inspected by a separate QoS module and then, based on the current node's state and the QoS requirements, the properties of the data packet are adjusted (see Figure 1).

Figure 1. 'Per-Packet Inspection' technique

For example, the priority of packets that have too much delay can be increased to ensure that the packet reaches its destination in due time. On the other hand, if a packet is near its destination and still has sufficient time left to fulfill its QoS requirements, its priority can be lowered so that less energy is used, or so that other packets with more stringent requirements arrive on time.

¹Complexity in this context means: the amount of computation, communication and decision making that is required by the technique.

Per-Packet Inspection can also be used to intelligently sort packets which are jitter sensitive, or even drop packets which are deemed less important or which will arrive too late anyway.

This technique requires only a local view on Quality of Service, both in time (only the current packet is considered) and space (this technique does not require communication with other nodes). As such, it is suited even for sensor nodes with very limited capabilities.

3.2 Protocol adaptation

A second QoS technique analyzes the inner workings of a protocol (such as routing or MAC) and, if needed, adjusts the settings and parameters of the protocol based on the QoS requirements (see Figure 2).

Figure 2. 'Protocol Adaptation' technique

For example, when a node notices that its queue is full, the node can decide to minimize the contention window of its MAC protocol to have a faster channel access and thus increase its throughput. Alternatively, more time slots can be requested. Similarly, the next-hop criteria of the routing protocol can be influenced depending on the QoS requirements. A low delay can be obtained by choosing next hops that result in the shortest path to the destination, whereas high reliability constraints might require a more nearby node with a lower packet loss rate.

This 'Protocol Adaptation' technique requires continuous interaction with arriving packets (both data and control packets), and as such results in a more global view in time. Although the other protocols on the node still communicate with other nodes, this technique requires only information of the current node and thus provides a very local view in space. More advanced nodes can make decisions based on a more global view in space by exchanging information between the QoS modules, at the cost of higher energy costs due to extra packet transmissions.

One of the major challenges for this technique is the definition of universal interfaces to retrieve and adjust protocol parameters.

3.3 Protocol replacement

Finally, instead of changing the way a protocol behaves, it is possible to replace network protocols by a new protocol. This new protocol can be more suited for the current network conditions and QoS requirements. As such, 'Protocol Replacement' requires a global view on the network situation both in time and space.

Protocol replacement can be applied at three different scales (see also Figure 3):

- *Node protocol replacement.* A single sensor node can decide to replace a network protocol. For example, a node can decide to change its data-aggregation protocol.
- *Local protocol replacement.* An alternative is 'local protocol replacement', whereby a group of sensor nodes changes one of their network protocols. For example, the North-East part of the sensor network may decide to replace their routing strategy from reactive to proactive.
- *Global protocol replacement.* Finally, when using 'global protocol replacement', the whole network decides to change one of their protocols. For example, because of lower network utilization, a CSMA-based MAC protocol may be more beneficial than a TDMA-based MAC protocol.

Figure 3. 'Protocol Replacement' technique

The use of a protocol replacement strategy has several consequences. First, the protocols must all be available on the node. This can be either in pre-compiled version, or through a code-exchange procedure. Furthermore, the architecture must support the switching of protocols: packets must be passed to the new protocol and the new protocol's parameters must be set in line with the current network conditions. Also, communication messages must be exchanged between the nodes, which introduces extra communication overhead. Finally, different protocols should be compatible, so that neighboring nodes running different protocols

Space	Time	Local	Global
	Local	Per-Packet Inspection	Protocol Adaptation
	Global	-	Protocol Replacement

Figure 4. QoS techniques and their network vision

can still communicate with each other. However, according to existing work [7], the protocol replacement approach can result in a substantial increase of the network performance.

In Figure 4, a table is given that indicates for each QoS technique whether a global or local network view (in terms of space and time) is required.

4 Adaptive Modular QoS Architecture (AMoQoS)

The previously presented QoS techniques are illustrated by integrating them in a conceptual architecture. This Adaptive Modular QoS Architecture (AMoQoS) is shown in Figure 5 and is designed to be integrated in a cross-layer or modular wireless sensor network architecture [10]. The AMoQoS is situated between an Application Abstraction Layer (AAL) and a Hardware Abstraction Layer (HAL).

The *Application Abstraction Layer* forms an abstraction of the different sensor network applications and provides a QoS translation method. The latter is responsible for the correct translation of the application QoS requirements to the network QoS requirements.

The *Hardware Abstraction Layer* is responsible for a seamless interaction between the AMoQoS and the different kinds of physical hardware.

Both the Application Abstraction Layer and the Hardware Abstraction Layer are beyond the scope of this research paper.

The Adaptive Modular QoS Architecture itself consists of four main building blocks (the 'QoS Framework', a 'Modular Database', the 'Network Protocols' and the 'Packet Queues'). These building blocks will each be discussed in more detail in the following sections, starting with the 'QoS Framework'.

The 'QoS Framework' contains several modules which, together, provide the required QoS guarantees. Some of these modules are optional and are marked with an asterisk. Only nodes with sufficient remaining energy will implement all the modules and thus all the functionalities, whereas nodes with a lower remaining energy will only implement a subset of these optional modules. This specification is described in the QoS profiles (see also section 2.1). Three modules ('Protocol Adaptation', 'Protocol Replace-

ment' and 'Per-Packet Inspection') have already been described in section 3.

4.1 Configuration module

The 'Configuration Module' is responsible for adjusting the settings of the network protocols. For this, a general interface is required through which these settings can be modified. A typical example parameter which can be adjusted is the contention window of the MAC protocol.

4.2 Control plane management module

When setting up the sensor network, a decision has to be made about *which* and *how many* nodes are provided with *which type* of QoS profile and *where* they must be placed (see also section 2.1). The 'Control Plane Management Module' is responsible for this organization and for providing all the QoS-related communication between the sensor nodes.

Organizing the QoS profiles is a continuous process. For instance, after the initial QoS profile organization, a node with an advanced QoS profile can decide to implement a more energy saving QoS profile because its remaining energy becomes too low. Another node will have to switch to an advanced QoS profile in order to still be able to deliver the requested QoS guarantees.

For the organization and the management of the QoS profiles of the different sensor nodes, a 'Control Plane' is set up above the traditional 'Data Plane' (see Figure 6). Each sensor node can exchange all its QoS-related information to its neighbors or to remote nodes by using this control plane. It is important to remark that only the QoS-related information needed for a correct QoS management is sent over this control plane. The data packets are sent across the data plane. Consequently, the communication links used for the data packets can be totally different from the communication links used for the control information. For example, the data packets can be routed by an optimized hierarchical routing protocol where only intermediate advanced nodes are next-hop candidates.

4.3 Information gathering module

The 'Information Gathering Module' has several responsibilities. First, it is responsible for answering requests for information from other modules. For instance, the 'Protocol Replacement Module' may need information about the remaining energy. Second, it is responsible for generating new requests for information to neighboring nodes when the requested information is not yet available (note that the actual distribution of these requests is the task of the 'Control Plane Management Module'). This can, for example,

Figure 5. Adaptive Modular QoS Architecture

be the load of the neighboring nodes. Finally, it is responsible for processing the (received) information and for storing it in the 'Modular Database'. For example, if the 'Protocol Adaptation Module' needs information about the load, this information must first be calculated. The 'Information Gathering Module' is then responsible for these calculations and for storing the results in the 'Modular Database'.

4.4 Activate/Deactivate module

When the 'Control Plane Management Module' has organized the nodes according to their QoS profiles, the 'Ac-

tivate/Deactivate Module' is responsible for activating and deactivating the corresponding code of the correct QoS profile in the 'QoS Framework'.

Furthermore, this module is responsible for activating and deactivating the code of the used network protocols. For example, when the 'Protocol Replacement Module' decides to change its routing protocol, this 'Activate/Deactivate Module' will deactivate the current routing protocol and activate the new routing protocol. When the sensor node does not contain the required protocol code, the 'Activate/Deactivate Module' will request the needed protocol code from other (neighboring) sensor nodes.

Figure 6. Overlay QoS Network

4.5 QoS management module

Central in the 'QoS Framework' is the 'QoS Management Module'. This module is responsible for the communication between the modules and for the communication with the 'Modular Database'.

A typical scenario goes as follows:

1. The 'Protocol Replacement Module' requires information about the remaining energy of the neighboring nodes.
2. This information is procured using the 'Information Gathering Module', who requests this information from the 'Modular Database'.
3. If this information is not yet available, the 'Information Gathering Module' generates a request for information from the neighboring nodes.
4. Finally, the 'Control Plane Management Module' is responsible for the correct distribution of this request.

This 'QoS Management Module' is thus responsible for the correct execution of the call sequences.

4.6 Data-aggregation module

The next module is the 'Data-aggregation Module'. Since data packets in sensor networks are often highly correlated, these packets can be often combined in order to reduce the number of packet transmissions [4]. Data-aggregation has a great influence on the QoS guarantees that can be delivered by the underlying sensor network.

- Since information from several packets is combined into a single packet at an intermediate node, data-aggregation has a negative influence on *reliability*. In

order to decrease the effect of packet losses, the maximal number of packets which are aggregated can be limited, or duplicates can be sent.

- Before aggregation can be executed, the node has to wait on the arrival of other data packets which should be combined. Thus, data-aggregation also has a negative influence on the introduced *delay*. To offer QoS guarantees, an upper limit on the waiting time can be imposed, or high-priority packets can be excluded from aggregation.
- Data-aggregation can have a profound influence on the *jitter* due to the introduced delay.
- Finally, data-aggregation can be used to reduce the traffic load. Since less packets have to be sent, the available *bandwidth* is more efficiently used.

4.7 Modular database

The 'Modular Database' is a central database that stores all the necessary code and parameters. A distinction can be made between the 'code' and the 'parameter' repository.

The *code repository* contains the code of the available network protocols and the code of the optional QoS Framework modules (the correct QoS profiles). The 'Activate/Deactivate Module' can control which code is active.

The *parameter repository* contains protocol, node, neighbor and network information, as well as the translation of the application requirements into QoS parameters. Every module that needs information about protocols, node or neighbors such as the contention window or the remaining energy, can (indirectly) interact with the 'Modular Database' to obtain or store this information.

4.8 Network protocols

The 'Network Protocols' building block contains all the active network protocols. For instance, two different routing protocols can be together active when there are multiple traffic streams each requiring a different routing strategy. Other typical protocols are a neighbor discovery protocol and MAC protocols.

4.9 Packet queues

Finally, the architecture contains an incoming and outgoing packet queue.

The incoming packet queue is used by the 'Per-Packet Inspection Module': the priority of incoming packets can dynamically be adapted and packets can be dropped. The queue itself classifies packets based on their priority. A weighted round robin system can be further implemented

to process the packets in the right sequence, and to meet the required QoS.

When a packet is fully processed, it is put into the outgoing packet queue. Afterwards, when the MAC module is ready to send a new packet, the outgoing QoS scheduler can intelligently select a packet from this queue and can send it to the MAC module.

5 Conclusion

In this paper, we investigated the need for adaptive QoS support in the next-generation wireless sensor networks and we proposed several solutions.

First, we described the dynamic and heterogeneous nature of these next-generation wireless sensor networks. This heterogeneity is expressed in two different ways: the nodes can have different capabilities and several applications can have many diverse requirements. Therefore, on one hand, we defined three different QoS profiles, and on the other hand, we presented several techniques that are able to dynamically adapt the behavior of the network. Some of these techniques such as 'Per-Packet Inspection' and 'Protocol Adaptation' can only influence the packet's or node's behavior, while others such as 'Local and Global Protocol Replacement' can influence the global network behavior.

Afterwards, we presented an Adaptive Modular QoS Framework (AMoQoS) in which the previous techniques and mechanisms can be combined. This framework is responsible for the continuous (re)organization of the QoS profiles, based on the current application requirements and on the remaining node capabilities.

Finally, we illustrated how this framework can dynamically adapt itself to the most optimal QoS configuration. For instance, different modules (e.g. routing) can be implemented at the same time in order to meet the requirements of different simultaneous traffic streams.

To conclude this paper, we are confident that QoS support will become increasingly important for the next-generation wireless sensor networks. However, current research does not take into account the heterogeneous nature of current sensor network hardware and applications. Therefore, we proposed several techniques and strategies which can be used to continuously deliver optimal QoS guarantees in dynamic and heterogeneous sensor networks.

References

- [1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury. A Survey on Wireless Multimedia Sensor Networks. *Computer Networks*, 51(4):921–960, 2007.
- [2] M. Caccamo, L. Y. Zhang, L. Sha, and G. Buttazzo. An Implicit Prioritized Access Protocol for Wireless Sensor Networks. In *Proceedings of the 23rd IEEE Real-Time Systems Symposium (RTSS'02)*, page 39, Washington, DC, USA, 2002. IEEE Computer Society.
- [3] D. Chen and P. K. Varshney. QoS Support in Wireless Sensor Networks: A Survey. In *Proceedings of the 2004 International Conference on Wireless Networks (ICWN 2004)*, Las Vegas, Nevada, USA, June 21–24 2004.
- [4] E. Fasolo, M. Rossi, and M. Widmer. J. and Zorzi. In-network Aggregation Techniques for Wireless Sensor Networks: A Survey. *Wireless Communications, IEEE [see also IEEE Personal Communications]*, 14(2):70–87, April 2007.
- [5] S. M.-E. Felemban, M.-C.-G. Lee, and M.-E. Ekici. MM-SPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks. *IEEE Transactions on Mobile Computing*, 5(6):738–754, 2006.
- [6] T. He, J. Stankovic, C. Lu, and T. Abdelzaher. SPEED: A Real-Time Routing Protocol for Sensor Networks. Report CS-2002-09, University of Virginia Tech., March 2002.
- [7] J. Hoebeke, I. Moerman, B. Dhoedt, and P. Demeester. Towards Adaptive Ad Hoc Network Routing. *International Journal of Wireless and Mobile Computing*, 1:1–8, 2005.
- [8] R. Jurdak, P. Baldi, and C. V. Lopes. Adaptive Low Power Listening for Wireless Sensor Networks. *IEEE Transactions on Mobile Computing*, 6(8):988–1004, 2007.
- [9] T. D. Little, J. Konrad, and P. Ishwar. A Wireless Video Sensor Network for Autonomous Coastal Sensing. In *Proceedings of the Conference on Coastal Environmental Sensing Networks (CESN 2007)*, Boston MA, April 2007.
- [10] E. D. Poorter, B. Latré, I. Moerman, and P. Demeester. Universal Modular Framework for Sensor Networks. In *International Workshop on Theoretical and Algorithmic Aspects of Sensor and Ad-hoc Networks (WTASA'07)*, Miami, USA, June 2007.
- [11] M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh. Exploiting Heterogeneity in Sensor Networks. In *Proceedings of the IEEE Infocom*, 2005.

The Second International Conference on Sensor Technologies and Applications

SENSORCOMM 2008

MESH 2008: International Conference on Advances in Mesh Networks

ENOPT 2008: The Second International Workshop on Energy Optimization in Wireless Sensors Networks

UNWAT 2008: The Second International Workshop on Under Water Sensors and Systems

25-31 August 2008

Cap Esterel, France

Editors

Mathilde Benveniste

Reinhardt Karnapke

Bart Braem

Jaime Lloret Mauri

Cosmin Dini

M. Sohrab H. Monsi

Giancarlo Fortino

Sponsored by

Published by

IEEE computer society

SENSORCOMM 2008 Information

Search

Papers by Session

Papers by Author

Getting Started

Trademarks

**The Second International
Conference on Sensor
Technologies and Applications
(SENSORCOMM 2008)**

IEEE
computer
society

MESM 2008: International
Conference on Advances
in Mesh Networks
ENOPT 2008: The Second
International Workshop
on Energy Optimization
in Wireless Sensors Networks
UNWAT 2008: The Second
International Workshop on Under
Water Sensors and Systems

**25-31 August 2008
Cap Esterel, France**

Editors

Mathilde Benveniste
Bart Braem
Cosmin Dini
Giancarlo Fortino
Reinhardt Karnapke
Jaime Lloret Mauri
M. Schrab H. Monel

Product Number E3330

BMS Part Number CFP0688C-CDR

ISBN 978-0-7695-3330-8

Library of Congress Number 2008928440

Copyright 2008 by The Institute of Electrical and Electronics Engineers, Inc. All rights reserved.