

## Measurement-Based Performance and Admission Control in Wireless Sensor Networks

Ibrahim Orhan

School of Technology and Health  
KTH  
Stockholm, Sweden  
Ibrahim.Orhan@sth.kth.se

Thomas Lindh

School of Technology and Health  
KTH  
Stockholm, Sweden  
Thomas.Lindh@sth.kth.se

**Abstract**—This journal paper presents a measurement-based performance management system for contention-based wireless sensor networks. Its main features are admission and performance control based on measurement data from lightweight performance meters in the endpoints. Test results show that admission and performance control improve the predictability and level of performance. The system can also be used as a tool for dimensioning and configuration of services in wireless sensor networks. Among the rapidly emerging services in wireless sensor networks we focus on healthcare applications.

**Keywords** - wireless sensor network, admission control, performance monitoring and control.

### I. INTRODUCTION

Wireless personal area networks have emerged as an important communication infrastructure in areas such as at-home healthcare and home automation, independent living and assistive technology, as well as sports and wellness. Initiatives towards interoperability and standardization are taken by several players e.g., in healthcare services. Zigbee Alliance has launched a profile for “Zigbee wireless sensor applications for health, wellness and fitness” [2]. The Continua Health Alliance promotes “an interoperable personal healthcare ecosystem” [3], and at-home health monitoring is also discussed in an informational Internet draft [4]. It shows that wireless personal area networks, including body sensor networks, are becoming more mature and are considered to be a realistic alternative as communication infrastructure for demanding services. However, to transmit data from e.g., an ECG in wireless networks is also a challenge, especially if multiple sensors compete for access as in CSMA/CA. Contention-based systems offer simplicity and utilization advantages, but the drawback is lack of predictable performance. Recipients of data sent in wireless sensor networks need to know whether they can trust the information or not. To address this problem we have developed a performance meter that can measure the performance [5], and furthermore, feed a performance control system with real-time measurement data [6]. This paper also discusses whether admission control in combination with a system for continuous performance management can provide improved and more predictable performance. Admission control is used in many traditional telecom systems. It is also proposed in new Internet service architectures [7] to provide guarantees for quality of service. In this paper we present a method for measurement-based admission control in wireless personal area sensor networks

for contention-based access. It is implemented as a part of an integrated performance management system that comprises performance monitoring, admission control and performance control.

The rest of the paper is organized as follows: a survey of related work in Section II; performance management in wireless sensor networks in Section III; measurement-based performance and admission control in Section IV; use cases and test results in Section V; and finally the conclusions in Section VI. This journal paper is an extension of a paper on admission control presented at a conference [1]. It provides a more detailed view of the other parts of the system, as well as the entire system for performance management.

### II. RELATED WORK

Performance in contention-based wireless networks using CSMA/CA has been studied extensively. Measurements, simulations and theoretical studies show that the loss ratio increases with the traffic load and number of sending nodes. Bianchi [8] has derived an analytical Markov chain model for saturated networks, further developed in [9] and extended to non-saturated networks in [10]. Channel errors due to e.g., external disturbances and obstacles in the environment, can of course increase the loss ratio further. Another related problem, studied in [11], is the reduced throughput in multi-hop networks, with one or several intermediate nodes between sender and receiver. Dunkels and Österlind [11] found that the implementation of packet copying in intermediate forwarding nodes has significant impact on the throughput.

Performance in low-rate WPAN has been analyzed in several simulation studies ([12] and [13]). A performance meter that keeps track of losses, inter-arrival jitter and throughput has been developed [5]. Several papers have also addressed congestion and rate control in WLAN and LR-WPAN. CODA (congestion detection and avoidance in sensor networks) is a control scheme that uses an open-loop backpressure mechanism as well as a closed-loop control, where a sink node can regulate a source node’s sending rate by varying the rate of acknowledgements sent to the source [14]. CARA (collision-aware rate adaptation) uses the RTS packets in IEEE 802.11 as probes to determine whether losses are caused by collisions (related to CSMA/CA) or by channel errors [15].

Our implementation of admission control, to accept or reject a request to join the network, is based on measurements of performance parameters, mainly the packet loss ratio. A

similar probe-based admission control procedure has been suggested for differentiated Internet services [7]. Alternatively, one can measure the available capacity between two endpoints, or on specific links in a network. Pathrate, Pathload and BART are examples of implementations of such estimation tools ([16], [17] and [18]). SenProbe [19] estimates the maximum achievable rate between two endpoints in wireless sensor networks by injecting packet trains and analyze the dispersion between the packets. Some experimental studies indicate that measurements of available capacity in wireless networks often are inaccurate, especially for multiple hops [20]. Instead of active measurements, the contention-aware admission control protocol (CACP) estimates the available capacity by letting each node measure the amount of time the channel is busy [21]. Perceptive admission control (PAC) is an extension of CACP to encompass node mobility [22]. We have preferred a straightforward approach where the decision to either accept or reject an admission request is based on direct measurements and estimates of the performance parameters that are decisive for the quality of services.

### III. PERFORMANCE MANAGEMENT IN WIRELESS SENSOR NETWORKS

A network scenario for the performance management system in this paper is depicted in Fig. 1. It consists of wearable sensors, such as ECGs, accelerometers, pulse-oximeters, fixed environment sensors, a coordinator, and intermediate nodes with routing and forwarding capabilities. An application program, running in the coordinator, processes sensor data from the sources and sends the information along with an estimate of the transmission quality to the remote end-user application for presentation and storage. The transmission quality can be expressed in terms of e.g., the statistical uncertainty of estimated parameters and the highest frequency component in a signal to be recovered by the receiver.

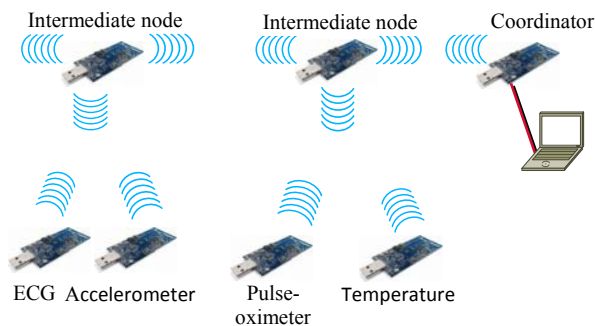


Figure 1. A network scenario where the performance management system is implemented in the coordinator and source nodes.

The performance monitoring and control capabilities can be implemented as add-on functions to be used by applications running in the communicating endpoints, e.g., sensor nodes and a coordinator, and not link by link. The ambition has also been to minimize the traffic overhead and energy consumption. The system is targeted to wireless sensor networks that use contention-based access, but can of course also be used in combination with contention-free access,

such as guaranteed time slots. The applications, e.g., streaming data from accelerometers and ECGs, require certain levels of throughput and a low loss ratio, however not necessarily zero. The aim is, firstly, to provide quality estimates of the transmitted parameters, and secondly, to reuse this information for admission and performance control of information loss, delays and throughput. This closes the loop between measurements and control.

Admission control needs to be seen in the context of other necessary functions, especially performance measurements and control. The performance manager consists of the following functions: a performance meter that collects measurement data; admission control that handles requests to join the network; and performance control that maintains the quality of service for the admitted sensor nodes. The performance meter provides feedback information for admission and performance control. Fig. 2 shows the relationship between these functions. A request from a sensor node to join the network is handled by the admission control based on feedback from the meter. The performance control function is responsible for maintaining the desired quality-of-service once the sensor nodes are allowed to use the wireless channel. The performance meter is described in the following subsection (III.A) and admission and performance control in Section IV.

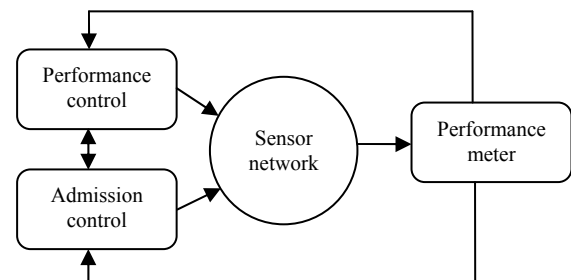


Figure 2. The performance manager consists of performance control and admission control. The performance meter supports the manager with measurement data.

#### A. Performance Meter

The approach is to combine active and passive techniques, inspired by the results from measurements in wired networks ([23] and [24]). A light-weight performance meter is implemented in each node. The meter consists of two counters that keep track of the number of sent and received packets and bytes, and a function that can inject monitoring packets. These dedicated measurement packets are inserted between blocks of ordinary data packets as seen in Fig. 3. They contain a sequence number, a timestamp and the cumulative number of packets and bytes transmitted from the sending node to the receiving node.

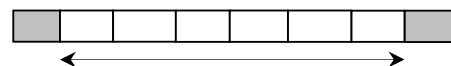


Figure 3. A monitoring block surrounded by two monitoring packets.

The interval between the monitoring packets, i.e. the size of the monitoring block, can be expressed in number of packets or a time interval, constant or varying randomly around a mean value. When a monitoring packet arrives, the receiving node stores a timestamp and the current cumulative counter values of the number of received packets and bytes from the sending node. Observe that for  $n$  sending nodes, the receiving node maintains  $n$  separate monitoring functions, one for each sending node.

Synchronization of the clocks in the participating nodes is not required. The local timestamps are used to calculate the inter-sending and inter-arrival times between pairs of monitoring packets. The inter-arrival jitter can then be calculated in a similar way as for RTP timestamps [25]. This means that the arrival time variation is estimated based on the monitoring packets, which represent samples of the ordinary data packet inter-arrival variation. Packet loss, on the other hand, is measured passively and directly using the counters.

### 1) Performance metrics

The following metrics can be calculated and estimated based on the collected measurements described in the previous subsection.

- Packet loss ratio: long-term average and average per monitoring block.
- The length of loss and loss-free periods defined as the number of consecutive monitoring blocks with or without losses. Can be expressed in time units, number of blocks, or number of packets and bytes.
- Inter-arrival jitter,  $J$ , is defined as  $J=(r_n-r_{n-1})-(s_n-s_{n-1})$ , where  $s$  is the sending time and  $r$  is the receiving time. The monitoring packets provide samples of this delay variation metric, which means that the uncertainty of the estimated statistics (mean value, median, percentiles etc.) is determined by the number of samples, and the variance of the delay process.
- Data throughput between sender and receiver can be calculated as a long-term average and also per monitoring block. The resolution of the peak rate is determined by the ratio between monitoring packets and ordinary data packets. This can also be seen as a measure of utilized capacity.

### 2) Meter and monitoring packet implementation

The performance meter is programmed in nesC [26] for TinyOS 2.1. The sensor nodes read samples from the sensors (ECG, accelerometer and temperature), assemble the samples and send them in packets to the coordinator (Fig. 4). The number of bytes and packets are counted. The cumulative number of bytes and packet and a timestamp are inserted into a monitoring packet, which is sent after every  $n$  ordinary data packet. A monitoring packet is 17 bytes long and includes the following fields: a start flag, a timestamp when packet is sent, type, a sequence number, number of packets sent, number of bytes sent, and a stop flag. The flags enable the coordinator to distinguish a monitoring packet from ordinary data packets. The sequence numbers identify and keep track of the monitoring packets. The packet and byte fields contain the cumulative number of bytes and packets sent. Finally, the

type field enables measuring several sensor data flows from the same node.

Each time the coordinator receives a data packet, it updates the number of bytes and packets received from each sensor. The coordinator uses the source field in the CC2420 radio header to distinguish the packets from different sources. When the coordinator receives a monitoring packet, it stores a timestamp and the cumulative counter values of the number of received packets and bytes from the sending node. Fig. 4 shows the measurement data sent from the performance meter to the performance manager. The table in the lower left part of Fig. 4 shows the information in each monitoring packet sent from a sensor node: a timestamp, the total number of bytes and the total number of packets sent from the sensor node. The table to the right shows the corresponding information added by the coordinator for each received monitoring packet.

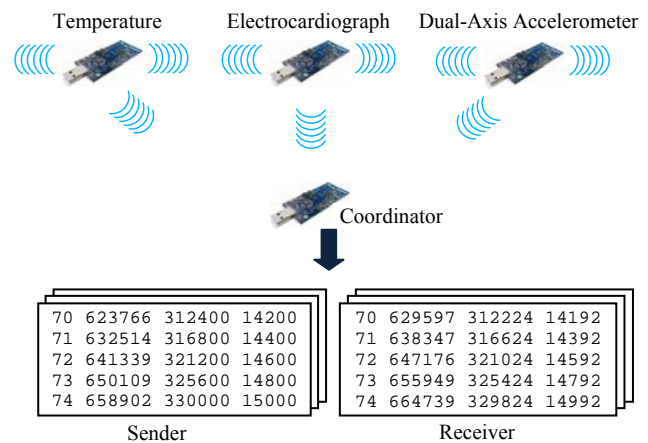


Figure 4. Measurement data from the sender and the receiver nodes. Columns from left to right: monitoring packet sequence no, timestamp (ms), cumulative number of bytes and packets.

## IV. ADMISSION AND PERFORMANCE CONTROL IN WIRELESS SENSOR NETWORKS

In this section the main idea behind the admission control (Section IV-A) and performance control (Section IV-B) system is presented. A star topology network controlled by a coordinator is used.

### A. Measurement-Based Admission Control for Contention-Based Access

A typical application scenario is healthcare at-home with a number of sensors, such as ECGs, pulse-oximeters, accelerometers etc., connected to a coordinator. Fig. 5 shows a scenario with three sensor nodes connected to a coordinator sharing the same wireless channel that applies the CSMA/CA access method. Several hops between the sensor nodes and the coordinator, as well as mobile sensor nodes, are also a feasible scenario. Sensor node A and sensor node B in Fig. 5 are already connected to the wireless channel transmitting sensor data to the coordinator. The sensor nodes have a specified throughput and an upper limit for the packet loss ratio. Sensor node C requests admission to join the network for a specified throughput and packet loss ratio.

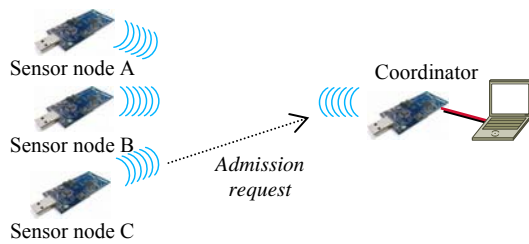


Figure 5. Sensor node C requests to share the wireless channel, already used by sensor node A and sensor node B.

The idea behind admission control is to accept or reject new sensor nodes to an existing network, while protecting the performance of already admitted nodes. Our purpose is to study whether it is feasible or not to use admission control in contention-based wireless sensor networks. The approach is to find the decision, to accept or reject an admission request, on estimates of real-time measurement data provided by a performance meter. A sensor node that intends to enter the network specifies the sampling rate, the sample size and the performance requirements. The verdict, to accept or reject the request, is determined by the outcome of probe packets transmitted during a test period. The probe packets sent from the requesting node to the coordinator should be of the same kind as the ordinary traffic it will transmit if admitted. The exchanged messages between a requesting node and the coordinator are described in the next subsection.

Strict performance guarantees are not feasible in contention-based access networks. However, many applications do not require completely loss-free transmission and are satisfied with soft performance requirements e.g., upper limits on packet loss and delay variation. The need for performance guarantees and predictability in contention-based networks for such applications is addressed in one of the use cases in Section V-C.

#### 1) Messages between the coordinator and sensor nodes

A simple protocol for exchange of messages between the coordinator and the sensor nodes have been defined (Fig. 6). Sensor nodes send requests to join the network for a specified sampling rate, sample size and upper limits on performance parameters. If the coordinator is not busy handling previous requests, it will approve further processing. The sensor node is then instructed to start transmitting probe packets interleaved by monitoring packets. When the test period ends, the sensor node asks the coordinator for the decision. Having received 'accept', the sensor node begins transmitting its ordinary data packets to the coordinator. Monitoring packets are inserted between blocks of  $n$  data packets or with certain time intervals, to provide the performance meter in the coordinator with real-time updates of the transmission quality.

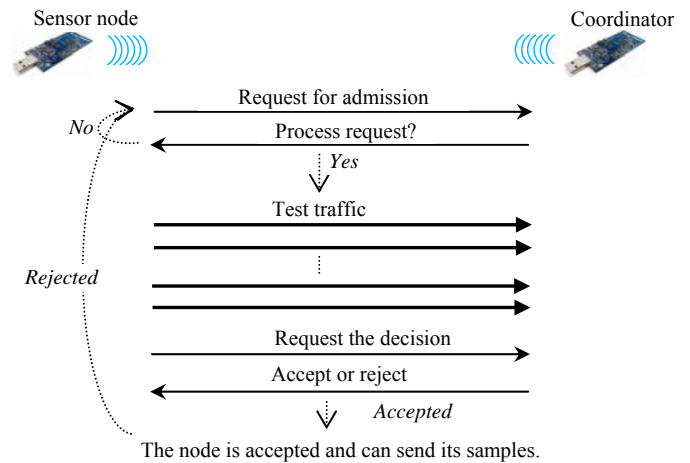


Figure 6. The messages between a sensor node and the coordinator during the admission phase. The arrows in thin lines are signalling messages and the arrows in bold lines represent probe packets in the test traffic phase.

#### 2) Admission test period

The sensor nodes transmit probe packets during the test period in the same way as they intend to do if the request is accepted. The performance meter will report performance data for traffic between the coordinator and all sensor nodes as well as the test traffic from the requesting sensor node. Admission is accepted if the averages of the performance parameters for any of the already permitted nodes, including the requesting node, are below the threshold value. Admission can be denied to protect the existing nodes from performance degradation. The length of the test period is a trade-off between retrieving enough information from the probe packets and minimizing the effect on the other sensor nodes' performance. The first priority is to protect the already admitted nodes. The test traffic phase will be interrupted as soon as the probe packets have the effect that e.g., the loss ratio threshold is exceeded. The probe packets sent during the test period can be seen as a sampling process of the wireless channel, where the outcome of each sampling event is that the packet is lost or succeeds. The probability to lose a packet depends on the total traffic load and the number of nodes that are transmitting (ignoring radio channel disturbances). The number of samples needed for a given confidence level is determined by the variance of the traffic load. We have assumed that the sampling frequencies of the sensors are stable. This is a reasonable assumption for the kind of the applications the system is intended for. It means that the variance of the traffic load over time is low, and accordingly, that the number of probe packets can be kept small. The experiences from the use cases (Section V-C) in a normal home environment confirm that a test period of less than 30 seconds is sufficient. The length of the test period is further discussed in Section V-C.

#### B. Performance Control System

The aim of the performance control system is, firstly, to provide quality estimates of the transmitted parameters, and secondly, to reuse this information for systems management and enable performance control in real-time e.g., to minimize

information loss and maintain a desired throughput. The output of the performance control system can also be to change the transmission power, enable or disable acknowledgement, etc. Applications, such as streaming data from accelerometers and ECGs in Fig. 7, require certain levels of throughput and a low loss ratio, however not necessarily zero.

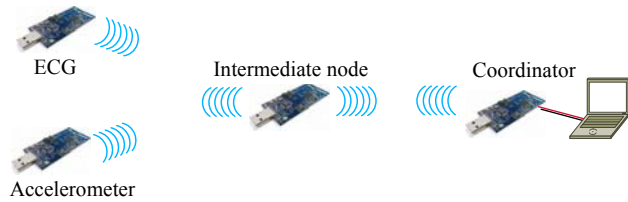


Figure 7. Two sensor nodes transmitting data to a coordinator via an intermediate node.

The performance control system (Fig. 8), implemented in a coordinator node, bases its decisions on the feedback information it receives from the meter e.g., packet loss, delays and throughput (packet loss is used in our cases). The meter delivers these performance updates for each incoming monitoring block e.g., once a second. The performance monitoring and control method has three main parameters. Firstly, the size of the monitoring block that determines the resolution of the performance metrics as well as the response time for the control actions. Secondly, the number of previous monitoring blocks ( $B_n$ ,  $B_{n-1}$ ,  $B_{n-2}$  etc), and their relative weight. The performance measurement results are calculated per each received monitoring block. To which degree the control method can rapidly adapt to changes is determined by these parameters. Thirdly, a step size ( $\Delta t$ ) controls the time interval between transmitted packets (and thereby the packet frequency). This step size determines the response time and also the stability of the system.

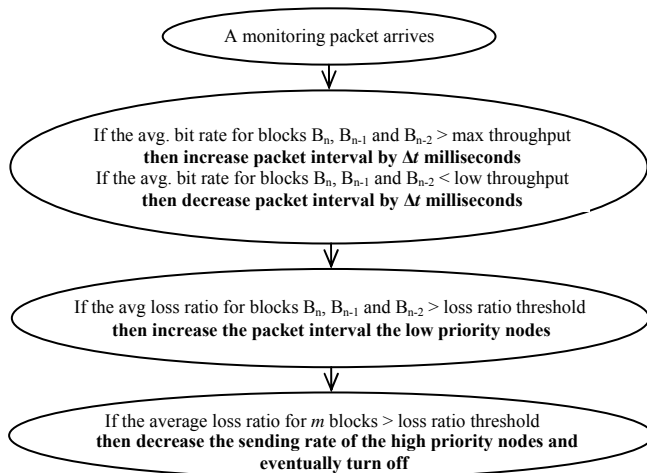


Figure 8. Flow diagram of the implemented control algorithm for prioritized nodes.

### 1) Algorithm to control throughput and loss ratio

The output of the control algorithm, to decrease or increase the packet frequency, is based on performance data from the current and previous monitoring blocks. The loss ratio and throughput (received bits per second) for a number of the recently received monitoring blocks are kept in memory. The manager sends a request message to a sensor node to either reduce or increase the packet frequency by adding (or subtracting)  $\Delta t$  milliseconds to (or from) the time interval between the transmitted packets.

### 2) Control algorithm with priority

A performance control system can support quality-of-service by assigning different priority to sensor nodes. Performance control with priority is primarily based on feedback information regarding packet loss and throughput from the respective source nodes. A use case with two levels of priority is described in more detail in Section V-B. High priority means that the required throughput (received bits per second) is maintained and the packet loss ratio is kept below a threshold for the prioritized nodes, possibly at the expense of nodes with low priority. If the loss ratio for the high-priority node is above the threshold, the manager will instruct the low-priority sensor nodes, to decrease their transmission rate step by step until the loss ratio for the high-priority node is below the threshold. If the loss ratio still is above the threshold, the sending rate of the high-priority nodes will be decreased as well, and eventually turned off if the loss ratio remains too high.

## V. USE CASES

In this section, we present use cases where the performance meter, performance control and admission control is used. Section V-A shows how the performance meter is used for online transmission quality feedback. Examples of parameters and statistical uncertainty are presented. Section V-B contains two cases: performance control to maintain throughput and keep packet loss below a threshold; and control with different and dynamically assigned priority. Section V-C illustrates the potential performance problems with contention-based access and the need for admission control, as well as continuing performance monitoring and control. The sensor node platform TmoteSky [28], running TinyOS 2.1 and programmed in nesC, is used in all cases below. The radio (CC2420) and link layer are compliant with IEEE 802.15.4 LR-WPAN [27] in contention-based access mode.

### A. Performance Meter – Online Transmission Quality Feedback

#### 1) The testbed and measurement scenarios

Two different network scenarios are studied. In Fig. 9 the sensor nodes are attached to the coordinator in a star topology.

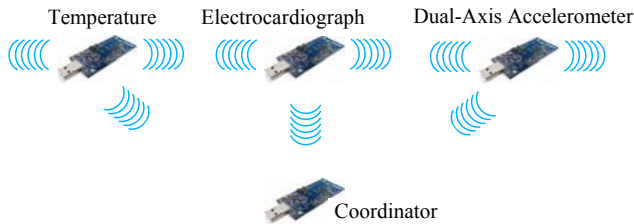


Figure 9. A network scenario with three sensor nodes and a coordinator.

In the second scenario (Fig.10), a sensor node is placed two hops away from the coordinator. The intermediate node forwards the packets from the sensor node to the coordinator. The buffer size in the intermediate node is 20 packets. In both scenarios, samples of sensor data are sent from the sensor node to the coordinator.



Figure 10. A network scenario with two hops between the sensor node and coordinator.

#### a) The temperature sensor

The temperature sensor in Fig. 9 is sampled twice a second and the collected samples are sent immediately to the coordinator. Monitoring packets are inserted between blocks of 100 data packets (6 byte payload).

#### b) The ECG sensor

One of the sensors in Fig.9 reads samples at 200Hz from the ADC-12 (analog-to-digital converters, 12 bits resolution) connected to an ECG. The samples are collected during five seconds. The radio is switched on, the samples are sent to the coordinator, and then switched off. This procedure is then repeated. Each packet contains 13 samples. The idea is to keep the radio turned off as long as possible and send several samples in each packet, in order to minimize the power consumption. In this case, the sensor node transmits 77 packets back-to-back every five seconds. A monitoring packet is inserted between blocks of approximately 100 ordinary data packets.

#### c) The dual-axis accelerometer sensor

A multi-sensor board (SBT80 from Easysen [29]) with a dual-axis accelerometer sensor is connected to two ADCs, one ADC for each axis. The accelerometer is sampled at 100Hz. The radio is only turned on during transmission. The sensor node sends 20 packets per second to the coordinator. Each packet carries 10 samples, 5 samples from each axis. Monitoring packets are inserted between blocks of 200 data packets, i.e. with approximately 10 seconds intervals.

### 2) Results and discussion

In this section some results using the performance meter in the two scenarios in Fig. 9 and Fig.10 are presented.

#### a) Loss periods and loss-free periods

The loss ratio per monitoring block during the measurement period for the accelerometer data is illustrated in Fig. 11. The distinct loss events in the beginning of the measurement period are caused by radio interferences. Table I shows the loss ratio per monitoring block and the mean length of loss periods and loss-free periods for the three wireless links in Fig. 9.

TABLE I. PACKET LOSS RATIO BETWEEN SENSOR NODES AND COORDINATOR

	Acc-Coord.	ECG-Coord.	Temp-Coord.
Mean loss ratio	0.038	0.002	0.006
Max loss ratio	0.935	0.040	0.100
Min loss ratio	0.000	0.000	0.000
Loss period mean length (s)	37s	6s	11s
Loss-free period mean length (s)	15s	40s	165s

The loss ratio during the three loss periods is between 0.8 and 0.9 (Fig. 11). The length of the loss-periods (consecutive monitoring blocks that contain at least one lost packet) is shown in Fig. 12.

#### b) Inter-arrival delay variation

Table II shows the inter-arrival delay variation (jitter) for the scenario in Fig. 10 with two hops between the sensor node and the coordinator compared to one hop. The sensor node transmits 20 packets per second. The radio communication is not exposed to disturbances in this case. Fig. 13 and Fig. 14 show that the inter-arrival jitter is several times higher with an intermediate node than without it. Packet loss for two hops is also considerably higher compared to one hop. The high levels of inter-arrival jitter and packet loss in the two-hop case is due to the intermediate node's receiving and forwarding capabilities.

TABLE II. INTER-ARRIVAL JITTER (MS)

Inter-arrival jitter (ms)	One hop	Two hops
Maximum	13	59
Minimum	0	0
Standard deviation	4.0	12.5

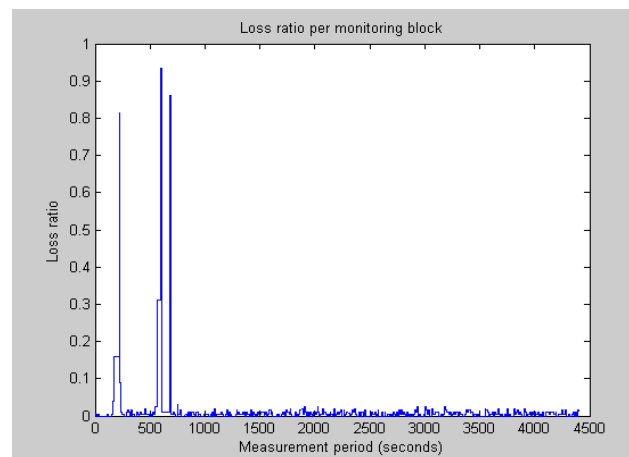


Figure 11. The loss ratio per monitoring block for accelerometer data in Fig. 10.

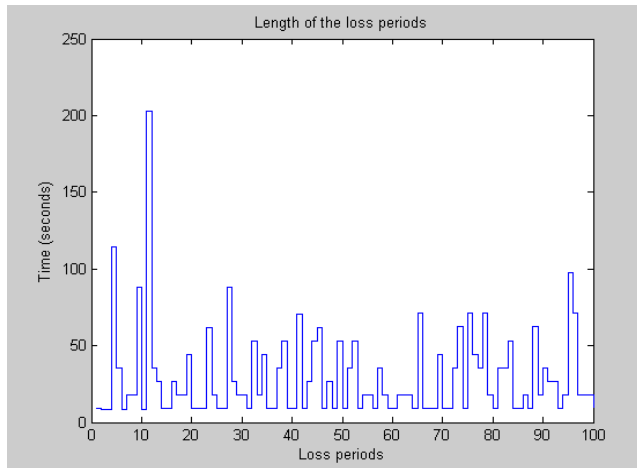


Figure 12. The length of loss periods in seconds.

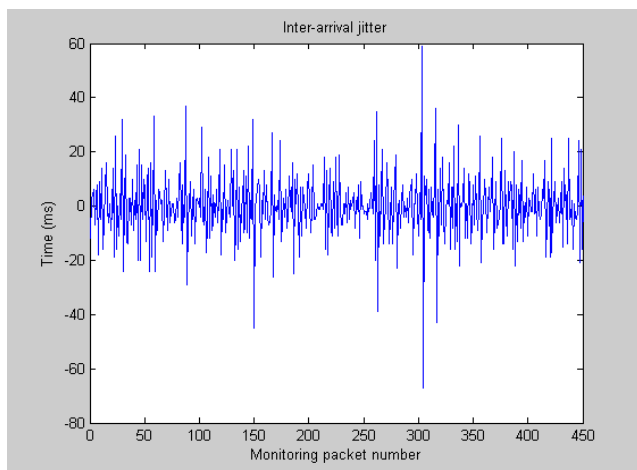


Figure 13. Inter-arrival jitter for a two-hop case.

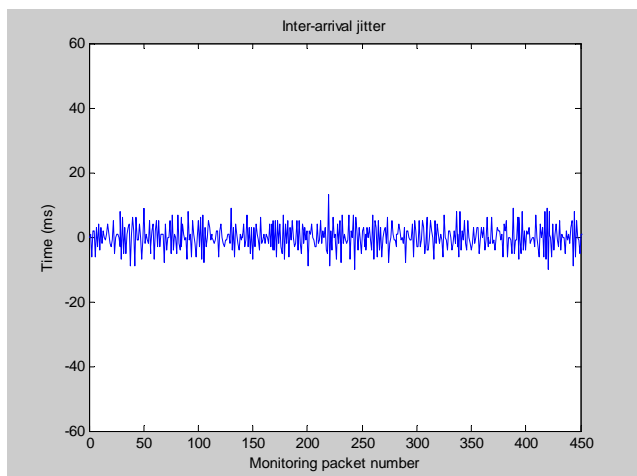


Figure 14. Inter-arrival jitter for a one-hop case.

### c) Uncertainty in parameter estimation

The result from the performance meter can be used for calculating the statistical uncertainty of the parameter estimates based on samples from sensors. Table III shows how

the confidence interval increases and highest frequency component in a received signal decreases when the loss ratio increases due to network performance degradation.

TABLE III. UNCERTAINTY IN ESTIMATION OF LOSS RATIO

Monitoring block duration (s)	Loss ratio	Conf. interval (0.99 level, stdev=4)	Highest frequency component in received signal
10s	0.025	0.93	50Hz
20s	0.313	1.11	35Hz
40s	0.935	3.62	3Hz
10s	0.010	0.93	50Hz
80s	0.861	2.47	18Hz
10s	0.030	0.94	50Hz
10s	0.005	0.93	50Hz
10s	0.005	0.93	50Hz
10s	0.010	0.93	50Hz

Details from the longest loss period in Fig. 12 are shown in Table III. The entire loss period consists of 20 monitoring blocks and lasts for 200 seconds. The monitoring block size in this case is 10 seconds. However, several blocks are longer e.g., the second, third and fifth row in Table III. The explanation is that monitoring packets, as well as data packets, may disappear before they arrive at the destination during a loss period. If one or several monitoring packets in a row are lost, the original monitoring blocks are merged into a larger block. Row 5 in Table III is a concatenation of 8 original blocks, where 7 monitoring packets were lost.

The loss ratio in Table III stretches from 0.005 to 0.935. The increased statistical uncertainty in estimating the mean value as the losses increase is shown in the third column. The standard deviation is around 4 units and the confidence level is chosen to be 0.99. The resulting confidence interval for an ideal communication channel without losses will be 0.92. A loss ratio of 0.313 (second row) leads to a confidence interval of 1.11, and a loss ratio of 0.935 gives a four times wider confidence interval (3.62). The number of samples,  $n$ , for a certain confidence interval,  $d$ , and confidence level ( $z=2.58$  for 0.99 confidence level), and standard deviation,  $s$ , is given

$$\text{by, } n = \frac{z^2 \cdot s^2}{(d/2)^2}.$$

The highest frequency component that can be recovered by the receiver for a 100Hz sampling rate is 50Hz (the sampling theorem). In this case the actual highest frequency component in the received signal is as low as 3Hz during the 36 seconds long period with a loss ratio of 0.935.

### B. Performance Control Algorithms

Three examples of performance control are presented in this section. The purpose of the first control algorithm is to maintain throughput and minimize losses for a node with high priority by punishing nodes with low priority (Section V-B.1). In the second case, where all nodes have the same priority (Section V-B.2), each node tries to maximize its throughput under the condition that the loss ratio is below a threshold. The third case (Section V-B.3) is a combination of the two previous ones. From the beginning both nodes have the same priority. After a certain time, one of the nodes is

dynamically assigned high priority and higher throughput. Finally, we show how the number of hops between a sensor node and the receiving coordinator determine the end-to-end throughput (Section V-B.4).

Fig. 15 shows the network scenario for the first case with two sensor nodes that are streaming ECG samples and accelerometer samples to the coordinator through a forwarding intermediate node.

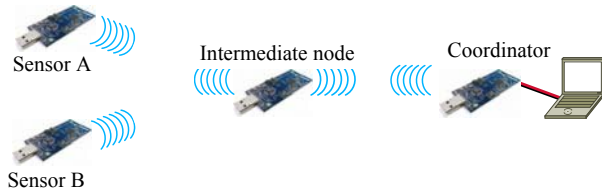


Figure 15. Two sensor nodes transmitting data to a coordinator via an intermediate node. Sensor node A has high priority and sensor node B has low priority.

### 1) Control with priority

The control algorithm in this case means that one of the sensor nodes is assigned high priority. The goal is to maintain throughput and keep loss ratio below an upper limit (0.02) for the high-priority node. The loss ratio threshold is computed as a weighted average of the three recent consecutive monitoring blocks and compared to the threshold 0.02. The required bit rate is 8kb/s, which corresponds to approximately 250Hz sampling rate per axis for a two-axis accelerometer or a 500Hz ECG.

Fig. 16 to Fig. 19 illustrate how the implemented algorithm works in practice. The high-priority node starts from 10kb/s and slows down to the expected bit rate 8kb/s (Fig. 16). The second node is turned on shortly thereafter ( $t \approx 80$ s) at a rate of nearly 16kb/s (Fig. 17). The received bit rate from the high-priority node falls sharply (Fig. 16). The solid lines (blue) show the received bit rate measured at the coordinator. The dotted lines (red) represent the sending bit rate from the sensor node. The loss ratio for the high-priority node peaks at almost 0.45 (Fig. 18), when the second node starts transmitting. The loss ratio for the low-priority nodes is shown in Fig. 18.

The performance manager reads the performance data provided by the meter for each block of incoming data packets. The monitoring block size is 100 packets in this test case. As soon as the manager detects the increased loss ratio for the high-priority node, it will instruct the other node to slow down. The low-priority node will directly decrease the transmitting rate (Fig. 17), which results in lower loss ratio (Fig. 18) and higher throughput (Fig. 15) for the prioritized node. As the loss ratio approaches the threshold, the sending rate of the low-priority node stabilizes around 3kb/s (Fig. 17). The performance manager strives to maintain the desired throughput (8kb/s) for the high-priority during the remaining part of the test, with an average loss ratio below the threshold.

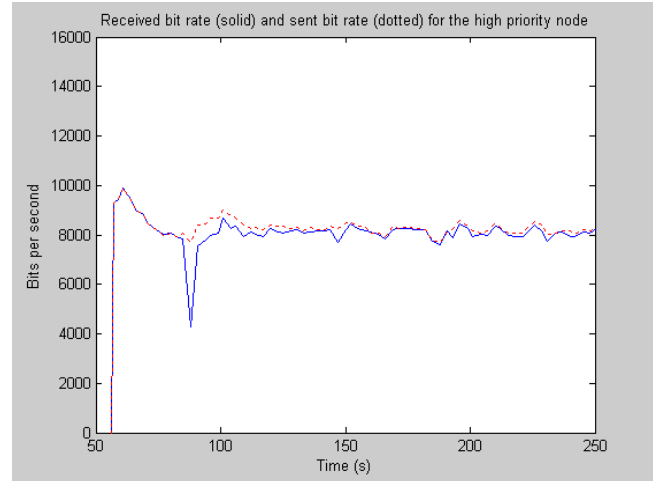


Figure 16. Throughput for the high-priority node.

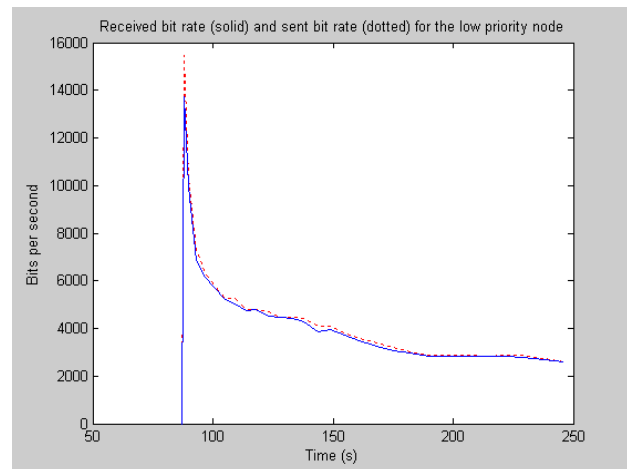


Figure 17. Throughput for the low-priority node.

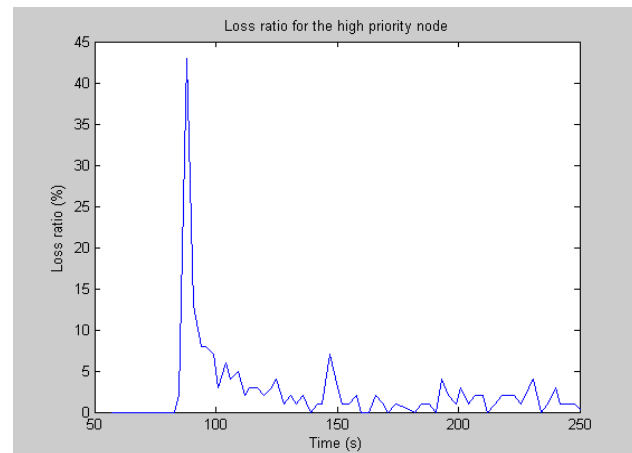


Figure 18. Loss ratio for the high-priority node.



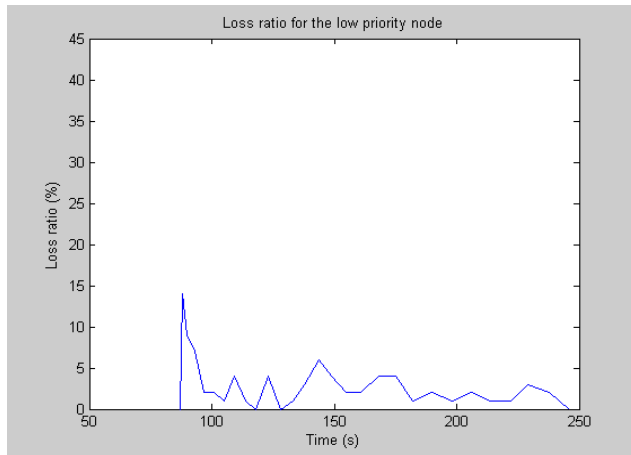


Figure 19. Loss ratio for the low-priority node.

### 2) Control without priority

In this case priority is not used. Both sensor nodes are controlled independently by the performance manager under the condition that the loss ratio is below a threshold. If the loss ratio exceeds the threshold, the sensor node will be instructed to decrease the sending rate (increase the packet interval by  $\Delta t$  milliseconds). No expected throughput is specified. Both sensor nodes start sending at 18kb/s as seen in Fig. 20 and Fig. 21. The high loss ratio for both nodes means that the performance manager will order both of them to slow down until the losses fall below the threshold. It can also be observed that the sensor node sometimes maintains the sending rate, even though the loss ratio is significantly higher than the threshold (Fig. 20 and Fig. 22). The explanation is that during heavy loss, monitoring packets will be lost as well, which delays the decision to decrease the packet frequency. After a while, the first node's throughput stabilizes around 3kb/s (Fig. 20) and around 3.5kb/s for the second node (Fig. 21). Since the control of the sensor nodes is independent of each other, the throughput will normally not be on the same level. One reason is different loss characteristics of the two channels; another may be different starting values. Each sensor node tries to find its maximum bit rate without exceeding the loss ratio threshold.

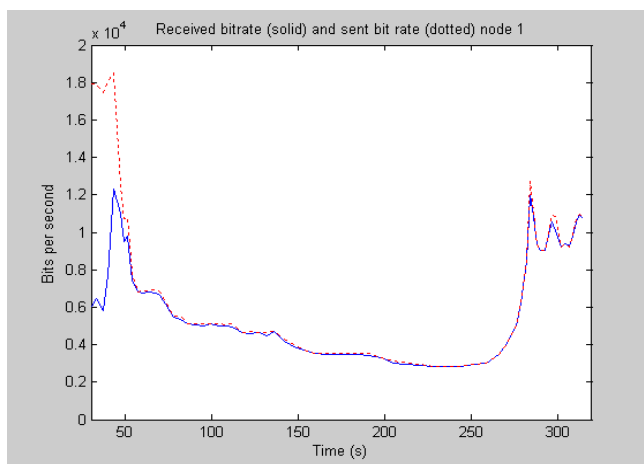


Figure 20. Throughput for node 1 (test case without priority).

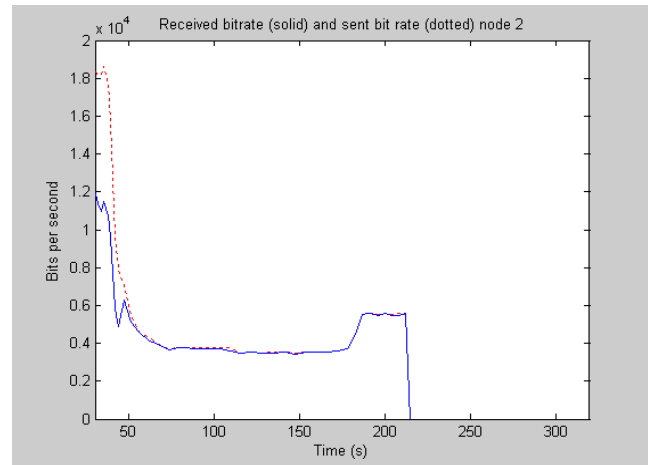


Figure 21. Throughput for node 2 (test case without priority).

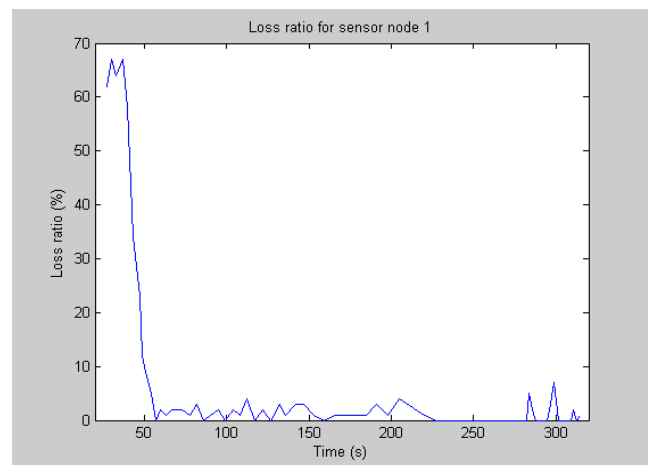


Figure 22. Loss ratio for node 1 (test case without priority).

At approximately  $t=180s$ , the manager has observed that the recent monitoring blocks are loss-free. The packet frequency is therefore increased for node 2 (Fig. 21). At  $t=210s$ , sensor node 2 stops transmitting (Fig. 21), which results in approximately zero packet loss for sensor node 1 (Fig. 22). The manager therefore tells the node to increase the packet frequency, up to around 10kb/s, where the loss threshold forces the node to slow down (Fig. 20).

### 3) Dynamic priority control

Fig. 23 and Fig. 24 show a combination of the previous two control algorithms. Both nodes start at a bit rate just below 15kb/s with 0.02 as the upper limit for the loss ratio. No node is given priority over the other. The throughput stabilizes between 4kb/s and 5kb/s. At  $t\approx 300$  seconds, one of the nodes (Fig. 15) is dynamically assigned high priority, whereas the other node has to be satisfied with what is left. The reason might be that a higher sampling rate is needed for a sensor. The bit rate for the high-priority node rises to the required 8kb/s (Fig. 23) and the other sensor node backs off to around 2.5kb/s (Fig. 24). The step response in Fig. 23 takes around 30s. This time period can be reduced either by allowing larger step sizes ( $\Delta t$ ) or decreasing the interval between the monitoring packets).

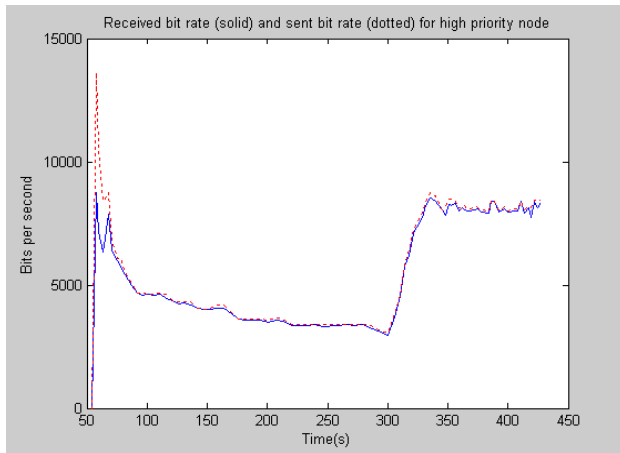


Figure 23. The sensor node is assigned high priority at  $t=300$ s and raises the bit rate to 8kb/s. The solid line represent received bit rate and the dotted line show sent bit rate.

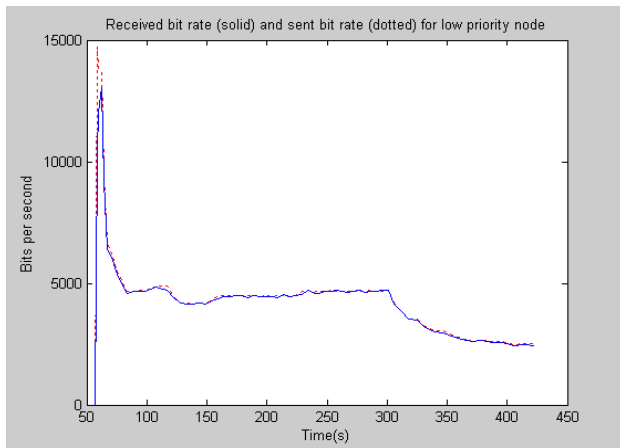


Figure 24. The sensor node is assigned low priority at  $t=300$ s and reduces the bit rate to 2.5kb/s. The solid line represent received bit rate and the dotted line show sent bit rate.

#### 4) Multi-hop cases

The bit rate from a sensor node to a coordinator will to a large extent depend on the number of hops between the source and destination [4]. We have measured throughput between a sensor node and the receiving coordinator for zero, one and two intermediate nodes. The maximum received throughput for the equipment in our testbed using maximum packet length (payload 112 byte) was 50kb/s for one hop, 35kb/s for two hops and 20kb/s for three hops. This is of course a crucial limitation for demanding applications.

#### 5) Results and discussion

Our analysis shows that it is feasible to use the measurement method, based on monitoring blocks, for performance monitoring as well as for feedback control of the performance of applications in wireless sensor networks. The results of the priority control algorithms are promising. The method has been implemented in a network with contention-based (CSMA/CA) access. It can of course also be used for the contention-based part of a super-frame in beacon mode in IEEE 802.15.4, where the contention-free part has guaranteed timeslots for the most demanding applications. One

observation is that it is more straightforward to avoid packet loss in situations of buffer saturation by reducing the packet frequency, than to handle packet loss due to collisions and channel errors.

The monitoring and control method has three main parameters, that can be tuned for optimal results: the size of the monitoring block ( $B$ ); the number of previous monitoring blocks ( $B_n$ ,  $B_{n-1}$ ,  $B_{n-2}$  etc) and their relative weight and, the step size ( $\Delta t$ ) that controls the time interval between transmitted packets (or packet frequency). A more systematic study of these aspects related to control theory is for future work. To find out, in real-time, what capacity is available for a specified loss ratio, given that a second node transmits at a certain bit rate, is another example of application for the performance control method.

#### C. Admission Control and Performance Issues

In the following section we present some of the potential performance problems with contention-based access and the need for admission control, as well as continuing performance monitoring and control. Section V-C.1 illustrates the non-trivial performance problems associated with contention-based access (CSMA). Section V-C.2 shows how admission control works in real-time. The length of the test period is also discussed. In the third case (Section V-C.3), the implemented system is used as an off-line configuration tool to determine how changes of the traffic pattern influence the packet loss ratio. Finally, the alternative to allocate a new radio channel to a requesting sensor node is mentioned in Section V-C.4. The testbed consists of sensor nodes transmitting samples from ECGs, pulse-oximeters and accelerometers with sampling rates from 100Hz to 250Hz to a coordinator.

##### 1) Performance problems in contention-based access

Contention-based access is a challenge for applications that require good and predictable performance. Fig. 25 illustrates what can happen when several sensors access a wireless channel. Three sensor nodes (A, B and C in Fig. 26) are connected to a coordinator sharing the same channel. The sensors are sampled during a second and the packets are sent back-to-back once a second. The bit rate is 9.6kbps for each sensor node. Fig. 25 shows the loss ratio during a measurement period for sensor node A. During the first part (0-70 seconds) only sensor node A is active. The loss ratio is almost zero. Between 70-140 seconds, sensor node B also accesses the channel. The average loss ratio experience by sensor node A is 0.03. During the remaining measurement period all three sensor nodes are transmitting on the same channel. The average loss ratio suddenly rises to 0.40.

For a loss-sensitive application, the performance is unacceptable after sensor node B, and especially after sensor node C, has joined the channel. The performance degradation may be avoided if the coordinator applies admission control and also maintains performance monitoring and control to protect the quality of service requirements for the existing nodes.

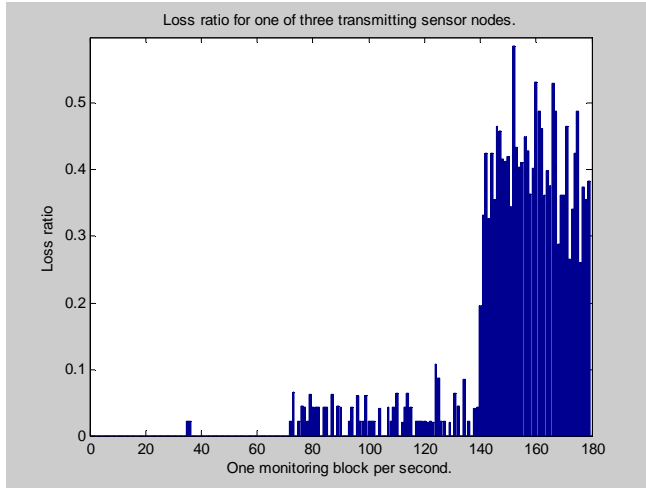


Figure 25. The loss ratio (y-axis) for sensor node A during the measurement period (x-axis in seconds). At approximately  $t=70s$  sensor B joins the channel. At  $t=140s$  a third node, sensor node C, accesses the channel.

## 2) Admission control

In this test case, the coordinator applies admission control when three sensor nodes with accelerometers, one by one, request to join the wireless network (Fig. 26). The sampling rate for the three-axis accelerometer is 200Hz per axis and the resulting average bit rate is 9.6kbps. The upper limit for packet loss for each node is set to 0.02 per monitoring block (the block length is around 1 second). The admission test period is 30 monitoring blocks (30 seconds). The measurement sequence is outlined in Fig. 27. Sensor node A requests admission and begins transmitting probe packets. The loss ratio during this admission test period is zero. Sensor node A's request is accepted and it starts transferring data. The loss ratio for the data traffic from sensor node A is almost zero before sensor node B requests to join the channel. Table IV summarizes the loss ratio for each sensor node during every test and data transfer period. Loss ratios exceeding the threshold (0.02) are indicated in bold text. It turns out that sensor node A and B are accepted, while sensor node C is rejected. For a sensor node to be rejected it is sufficient that the loss ratio for one of the sensor nodes, including the requesting node itself, exceeds the threshold.

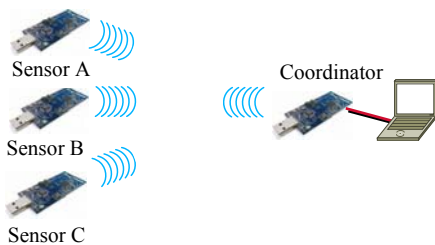


Figure 26. Three sensor nodes connected to the coordinator sharing the same channel.

The length of the test period is a trade-off between, on the one hand, to minimize the disturbance of existing traffic and reducing the response time for the admission verdict, and on the other hand, to receive sufficient performance data.

The drawback of a predetermined fixed length of the test period is that ongoing traffic may suffer from severe performance deterioration. Fig. 28 shows the impact of test traffic on a sensor node during a 30 seconds test period. The average loss ratio is almost 0.05, with several peaks around 0.10, which is unacceptable performance deterioration for an already admitted node during a test period. To avoid this, we use an algorithm that calculates the cumulative moving average of the loss ratio for each incoming performance update i.e., for each monitoring packet. The test period is interrupted if the cumulative average exceeds a threshold. The cumulative moving average is defined as  $CA_i=(L_1+L_2+L_3+\dots+L_i)/i$ , where  $L_i$  is the loss ratio for monitoring block  $i$ . The algorithm is applied to the three test periods in Fig. 28 – Fig. 30. The cumulative averages for the first five blocks in Fig. 28 are  $CA_1=0.059$ ,  $CA_2=0.035$ ,  $CA_3=0.032$ ,  $CA_4=0.042$  and  $CA_5=0.042$ .

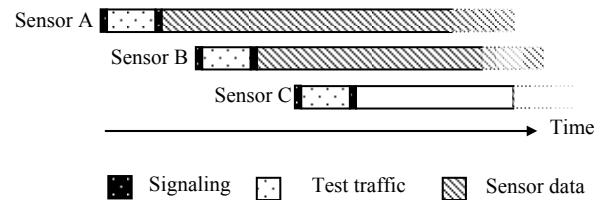


Figure 27. The measurement sequence for the nodes in Fig.26.

TABLE IV. LOSS RATIO FOR TEST PERIODS AND DATA TRANSFER PERIODS FOR SENSOR NODE A, B AND C.

	Sensor A	Sensor B	Sensor C
Test period sensor A	0.0000	--	--
Data transfer	0.0006	--	--
Test period sensor B	0.0012	0.0085	--
Data transfer	0.0019	0.0088	--
Test period sensor C	0.0046	<b>0.0470</b>	<b>0.0250</b>
Data transfer	0.0051	0.0083	--

If the rule for admittance is to allow maximum three consecutive updates of the loss ratio above the threshold (0.02), the test period will be interrupted after the third block. With an additional requirement that the loss ratio for a single block cannot exceed 0.05, this example means that the test period is interrupted after the first monitoring block.

A slightly different loss pattern is depicted in Fig. 29 (sensor node C's loss ratio during a test period). The cumulative average for the first seven blocks are  $CA_1=0.0118$ ,  $CA_2=0.0119$ ,  $CA_3=0.0159$ ,  $CA_4=0.0240$  and  $CA_5=0.0216$ ,  $CA_6=0.0201$  and  $CA_7=0.0206$ . In this case, the test period terminates after the 6<sup>th</sup> monitoring block and the request is rejected.

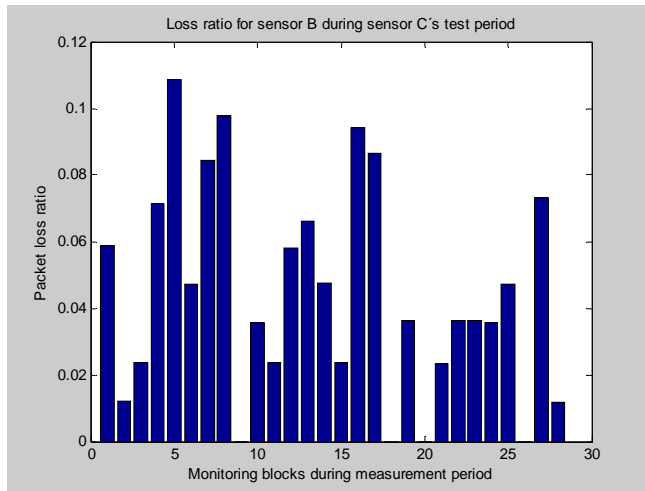


Figure 28. Loss ratio per monitoring block experienced by sensor node B during the third sensor's (sensor node C) test period. The average loss ratio is 0.047.

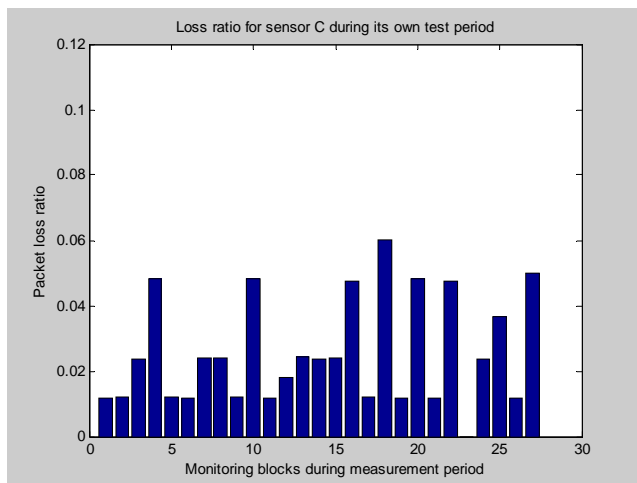


Figure 29. Loss ratio per monitoring block experienced by sensor node C during its own test period. The average loss ratio is 0.025.

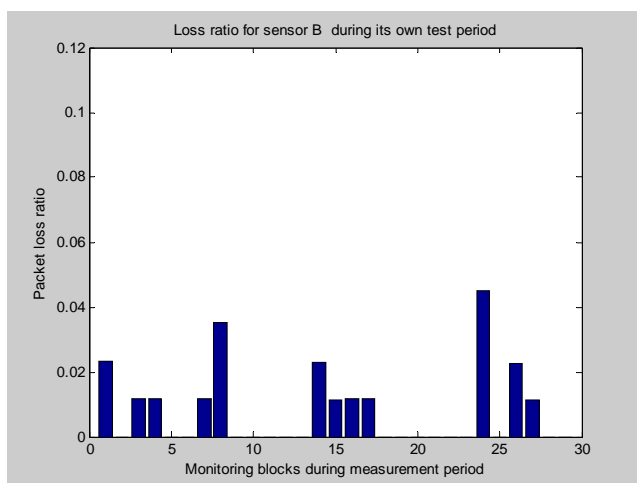


Figure 30. Loss ratio per monitoring block experienced by sensor node B during its own test period.

### 3) Traffic patterns and channel access

Packet loss in contention-based wireless networks is sensitive to the traffic pattern from the individual sources. Assume that two nodes collect samples and transmit the samples as a train of packets periodically once a second. If the nodes transmit the packet trains without overlap in time, the risk for losses due to collisions is low. However, the loss probability will increase if the packet trains happen to coincide. The dynamics of the traffic patterns in a network may from time to time lead to losses that exceed the accepted level after the admission test periods. The unpredictability of performance deterioration in wireless contention-based networks means that admission control must be combined with continuous traffic monitoring and control to be able to maintain the desired performance goals.

We have performed tests to study the impact of changes in traffic pattern on packet loss. Sensor node A collects and stores samples during a second. The samples are encapsulated in packets and transmitted back-to-back. The total time to transmit the packet train depends on the sampling rate, the sample size and the packet size. In this case, the sensor node sends a packet train of 43 packets with a packet size of 28 bytes, which corresponds to a throughput of 9.6kb/s. The total time to send the packet train was around 500ms. A second node, sensor node B, starts transmitting probe packets. It sends a train of packets once a second during the test period. The starting time for each train is shifted 50ms after ten seconds. This is repeated ten times, which means that the total time shift of the packet trains is around 500ms. The basic idea is to let the packet trains from sensor node B slide over the packet trains from sensor node A. Fig. 31 illustrates this convolution-like procedure.

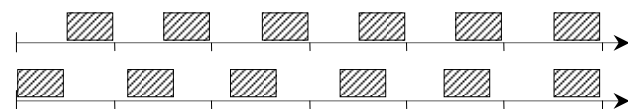


Figure 31. Sensor node A (the upper part) sends packet trains periodically every second. The starting times of the trains transmitted by sensor node B (lower part) are shifted in time so that they slide over the packet trains from sensor A.

Fig. 32 shows the loss ratio for sensor node B. After 10 monitoring blocks (10 seconds), the starting time is shifted 50ms. The average loss ratio for the first half of the measurement period is below 0.01. It rises to 0.10 for block 81-90 and 0.17 for block 91-100. The highest losses occur when the packet trains from the two sensors coincide in time. This convolution-like test might be inappropriate to use in an operating network but is useful for out-of-service configuration and dimensioning tests to estimate a worst case loss ratio. The traffic pattern for a channel e.g., the starting times of packet trains, is a stochastic process that may result in random losses from zero up to 0.25 in this case. Due to the unpredictability of contention-based wireless access continuous performance monitoring and control is needed to maintain the desired performance levels.

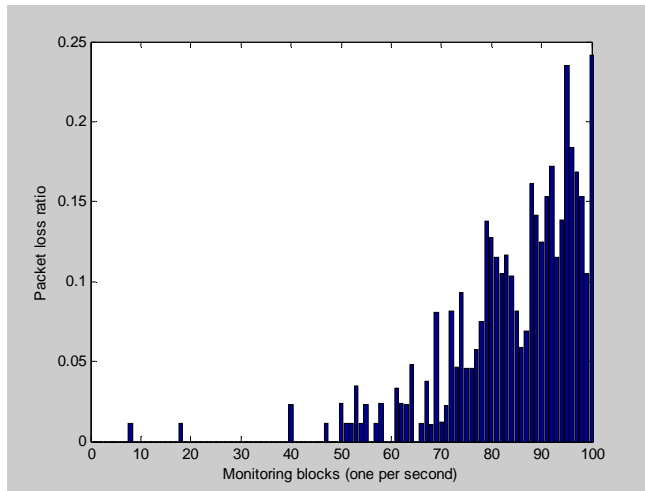


Figure 32. Loss ratio for sensor node B. The peak values occur when the packet trains from sensor node B coincide in time with the packet trains from sensor node A.

#### 4) Redirecting to another channel

When a sensor node's request to join the network is rejected there are two alternatives. The node may back off for a while and try once again later. Alternatively, the coordinator may refer the sensor node to another radio channel. This feature has been successfully implemented and tested.

#### 5) Results and discussion

The length of the test period is a trade-off between minimizing the disturbances on existing traffic, and receiving sufficient performance data for the admission verdict. The proposed algorithm uses a cumulative moving average of the loss ratio for the traffic from each sensor node to decide whether to reject an admission request and interrupt the test traffic, or to permit the sensor node to use the network. The test results show that admission control can improve the level, and predictability, of the performance of wireless sensor nodes. In addition, the method is also suitable for dimensioning, configuration and testing prior to operational mode. It can determine the number of sensor nodes that can share a wireless channel, for given performance requirements. A final conclusion is that continuous performance monitoring and control is needed to maintain the desired performance levels.

## VI. CONCLUSIONS

Wireless sensor networks have today emerged as a feasible infrastructure for demanding applications e.g., in health-care. This paper has addressed the non-trivial performance problems related to contention-based access to wireless channels. We have presented a measurement-based system for admission and performance control in wireless sensor networks. The measurements are provided by a distributed light-weight performance meter. The test result shows that the implemented admission and performance control functions improve the quality, and predictability, of demanding services. The system can also be used as a tool for dimensioning and configuration of services in wireless sensor networks.

## REFERENCES

- [1] T. Lindh and I. Orhan, "Measurement-Based Admission Control in Wireless Sensor Networks", *Sensorcomm*, pp. 426-431, Venice/Mestre, July 2010.
- [2] "Zigbee wireless sensor applications for health, wellness and fitness", Zigbee Alliance, March 2009.
- [3] R. Carroll, R. Cnossen, M. Schnell, and D. Simons, "Continua: an Interoperable Personal Healthcare Ecosystem", *IEEE Pervasive Computing*, Vol. 6, No. 4, October-December 2007.
- [4] A. Brandt (Zensys Inc) and G. Porcu (Telecom Italia), "Home Automation Routing Requirements in Low Power and Lossy Networks", Internet Draft, September 2009.
- [5] I. Orhan, A. Gonga, and T. Lindh, "An End-to-End Performance Meter for Applications in Wireless Body Sensor Networks", *BSN 2008*, pp. 330-333, Hongkong, June 2008.
- [6] T. Lindh and I. Orhan, "Performance Monitoring and Control in Contention-Based Wireless Sensor Networks", *International Symposium on Wireless Communication Systems*, pp. 507-511, Siena, Italy, September 2009.
- [7] I. Más and G. Karlsson, "Probe-based admission control for differentiated-services internet", *Computer Networks* 51, pp.3902-3918, September 2007.
- [8] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function", *IEEE JSAC*, Volume 18, No 3, pp 535 - 547 March 2000.
- [9] Hai L. Vu, "Collision Probability in Saturated IEEE 802.11 Networks", *Australian Telecommunication Networks and Applications Conference*, pp. 21-25, Australia, December 2006.
- [10] K. Duffy, D. Malone, and D.J. Leith, "Modeling the 802.11 Distributed Coordination Function in Non-saturated Conditions", *Communications Letters, IEEE*, Volume 9, Issue 8, pp. 715-717, August 2005.
- [11] F. Österlind and A. Dunkels, "Approaching the Maximum 802.15.4 Multi-hop Throughput", *HotEmnets*, Virginia, June 2008.
- [12] D. Cavalcanti et al, "Performance Analysis of 802.15.4 and 802.11e for Body Sensor Network Applications", *BSN*, Aachen, March 2007.
- [13] N. Golmie et al: "Performance analysis of low rate wireless technologies for medical applications" *Computer Communications*, Volume 28, Issue 10, pp 1266-1275, June 2005.
- [14] C.Y. Wan, S.B. Eisenman, and A.T. Campbell, "CODA: congestion detection and avoidance in sensor networks", *SenSys*, 1st conference on embedded networked sensor systems, pp 266-279, Los Angeles, November 2003.
- [15] J. Kim, S. Kim, S. Choi, and D. Qiao, "CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs", *INFOCOM*, pp 1-11, Barcelona, April 2006.
- [16] P. Ramanathan, D. Moore, and C. Dovrolis "What Do Packet Dispersion Techniques Measure", In *Proceedings of IEEE INFOCOM*, 2001, pp. 905-914, Anchorage, Alaska, USA, April 2001.
- [17] M. Jain and C. Dovrolis, "Pathload: a measurement tool for end-to-end available bandwidth", *Passive and Active Measurements Workshop*, pp 14-25, Fort Collins, USA, March 2002.
- [18] S. Ekelin, M. Nilsson, E. Hartikainen, A. Johnsson, J-E Mångs, B. Melander, and M. Björkman, "Real-Time Measurement of End-to-End Available Bandwidth using Kalman Filtering", *IEEE NOMS*, pp 73-84, Vancouver, Canada, April 2006.
- [19] T. Sun, L. Chen, G. Yang, M. Y. Sanadidi, and M. Gerla, "SenProbe: Path Capacity Estimation in Wireless Sensor Networks" *SenMetrics 2005*, San Diego, USA, July 2005
- [20] D. Gupta, D. Wu, P. Mohapatra, and C-N. Chuah, "Experimental Comparison of Bandwidth Estimation Tools for Wireless Mesh Networks", *IEEE INFOCOM Mini-Conference*, pp 2891- 2895, April 2009.

- [21] Y. Yang and R Kravets, "Contention-Aware Admission Control for Ad Hoc Networks" *Mobile Computing, IEEE Transactions on* Volume 4, Issue 4, pp 363 - 377, July-August, 2005.
- [22] Ian D. Chakeres and Elizabeth M. Belding-Royer, "PAC: Perceptive Admission Control for Mobile Wireless Networks", *International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine)*, pp 18-26, Dallas, USA, October 2004
- [23] T. Lindh and N. Brownlee: "Integrating Active Methods and Flow Meters - an implementation using NeTraMet", *Passive and Active Measurement workshop (PAM2003)*, San Diego, April 2003.
- [24] M. Brenning, B. Olander, I. Orhan, J. Wennberg, and T. Lindh: "NeTraWeb: a Web-Based Traffic Flow Performance Meter", *SNCNW2006*, Luleå, Sweden, October 2006.
- [25] "RTP: A Transport Protocol for Real-Time Applications", RFC 3550, H. Schulzrinne et al., July 2003.
- [26] D. Gay, P. Lewis, R. von Behren, M. Welsh, E. Brewer, and D. Culler, "The nesC language: A holistic approach to networked embedded systems", *Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation*, pp 1-11, San Diego, USA, June 2003.
- [27] IEEE Standard 802.15.4 - 2006.
- [28] Tmote Sky – IEEE 802.15.4 compliant sensor module from Sentilla (previously Moteiv).
- [29] STB80 – Multi-Modality Sensor Board for TelosB Mote, [www.easysen.com](http://www.easysen.com), May 2011.