Cross-layer Quality of Service Support for UWB Wireless Multimedia Sensor Networks

Tommaso Melodia
Department of Electrical Engineering
State University of New York at Buffalo
Buffalo, NY 14260
tmelodia@eng.buffalo.edu

Ian F. Akyildiz
Broadband Wireless Networking Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332
ian@ece.gatech.edu

Abstract—**Wireless Multimedia Sensor Networks (WMSNs) are networks of wirelessly interconnected devices that allow retrieving video and audio streams, still images, and scalar sensor data. WMSN require the sensor network paradigm to be re-thought in view of the need for mechanisms to deliver multimedia content with a pre-defined level of quality of service (QoS). In this paper, a new cross-layer communication architecture based on the time-hopping impulse radio ultra wide band technology is described, designed to reliably and flexibly *deliver QoS to heterogeneous applications in WMSNs*, by leveraging and controlling interactions among different layers of the protocol stack according to applications requirements. Simulations show that the proposed system achieves the performance objectives of WMSNs without sacrificing on design modularity.**

I. Introduction

The availability of inexpensive hardware such as CMOS cameras and microphones that can ubiquitously capture multimedia content from the environment has fostered the development of Wireless Multimedia Sensor Networks (WMSNs) [1], i.e., networks of wirelessly interconnected devices that can retrieve video and audio streams, still images, and scalar sensor data. WMSN will enable new applications such as multimedia surveillance, traffic enforcement and control systems, advanced health care delivery, structural health monitoring, and industrial process control.

Existing sensor networks are mostly based on variants of the *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) Medium Access Control (MAC) protocol. CSMA/CA has demonstrated to be an effective mechanism to distributively share a common wireless channel among uncoordinated devices. However, it requires mutually exclusive transmissions, i.e., when a device is receiving data, transmissions from all the devices in its transmission range are impeded. Mutual exclusion is achieved by distributively coordinating the transmissions of different sensors mainly by means of two mechanisms, i.e., *carrier sense* and *random timers* to defer transmissions. While random timers lead to variable and uncontrollable access delays, carrier sense causes consistent energy consumption for idle listening; still, frequent collisions occur due for example to the *hidden node* problem, in turn leading to increased energy consumption and delays.

For the reasons above, although recent proposals [2][3] have modified existing protocols based on CSMA/CA and geo-graphical routing to provide delay-sensitive and error-resilient services in sensor networks, we believe that the application requirements of WMSNs call for a new design perspective and next-generation wireless technologies. Hence, in this paper we describe the preliminary design of a new cross-layer communication architecture to reliably and flexibly *deliver QoS to heterogeneous applications in WMSNs*, by leveraging and controlling interactions among functionalities handled at different layers according to applications requirements. Our design is based on the Time-Hopping Impulse Radio UWB (TH-IR-UWB) transmission technique.

The remainder of the paper is organized as follows. In Section II we outline the main design principles, and describe the proposed cross-layer architecture. In Section III, we describe the routing and admission control functionalities. In Section IV we describe the medium access control and the proposed dynamic code assignment and scheduling policies. In Section V we discuss performance evaluation results while in Section VI we draw the main conclusions.

II. Design Principles and Cross-layer Controller

The main design principles are as follows:

- **Network Layer QoS Support enforced by a cross-layer controller.** The proposed system provides QoS support at the network layer, i.e., it provides packet-level service differentiation in terms of throughput, end-to-end packet error rate, and delay. The cross-layer control unit (XLCU) configures and controls the networking functionalities at the physical, MAC, and network layer, based on a unified logic that takes decisions based on i) application requirements specified by the application layer; ii) the status of the functional blocks implementing the networking functionalities.
- **Hop-by-Hop QoS contracts.** End-to-end QoS requirements are enforced through local interactions. Each device is responsible for locally guaranteeing given performance objectives. The global, end-to-end requirement is thus guaranteed by the joint local decisions of the participating devices, as further explained in Section III. Note that time-based approaches in UWB allow ranging accuracy in the order of centimeters. Hence, our module

leverages geographical information to provide QoS, as further explained in Section III.

- **Receiver-centric scheduling for QoS Traffic.** In multi-hop wireless environments interference is location-dependent. For this reason, we provide QoS through receiver-centric scheduling. The receiver can be responsive to channel dynamics based on local measurements and consequently control loss recovery and rate adaptation, thus avoiding feedback overheads and latency.

- **UWB Physical/MAC layer.** We rely on an integrated MAC and physical layer based on UWB. Like CDMA, TH-IR-UWB allows multiple transmissions in parallel. This allows devising MAC protocols with minimal coordination, as will be discussed in Section IV.

- **Dynamic Channel Coding.** Power control is not beneficial in TH-IR-UWB [4]. Hence, adaptation to interference at the receiver is achieved through dynamic channel coding, which can be seen as an alternative form of power control, as it modulates the energy per bit according to the interference perceived at the receiver [4]. The proposed system includes a *channel encoder* block that adds redundancy to combat channel impairments and multi-user interference. The encoder at node i receives a block of L uncoded bits, selects the encoding rate $R_{E,i}$, which represents the number of data bits per encoded bit, among the set $\underline{R}_E = [R_E^1, R_E^2, \cdots, R_E^P]$, with $R_{E,i}^1 = 1$ (i.e., transmitting uncoded data), and with $R_E^1 > R_E^2 > \cdots > R_E^P$.

- **Different Traffic Classes.** WMSNs will need to provide support and differentiated service for several different classes of applications. The requirements of an application A are described as a set of tuples $\Psi^A = \{\psi^a(\delta^a, \beta^a, \zeta^a), a \in 1, \dots, N_\psi^A\}$. Here, ψ^a, $a \in 1, \dots, N_\psi^A$ represent N_ψ^A different subflows of the flow generated by application A. For each subflow ψ^a, δ^a represents the maximum allowed end-to-end delay for packets associated with the subflow, β^a represents the required bandwidth, and ζ^a indicates the end-to-end packet error rate (PER) that can be sustained by the subflow. A *QoS Adapter* block can split an application flow into several *subflows* each with defined characteristics. In the remainder of the paper we consider application flows at the level of subflows, i.e., a QoS adapter generates flows with characteristics $\psi^a(\delta^a, \beta^a, \zeta^a)$.

III. DISTRIBUTED ADMISSION CONTROL FUNCTIONALITY

Let us consider a flow $\psi^a(\delta^a, \beta^a, \zeta^a)$ generated at node i that requires service. A multi-hop path from i to the destination N needs to be established, with maximum end-to-end delay δ^a, minimum guaranteed bandwidth β^a, and maximum end-to-end packet error rate ζ^a.

The required bandwidth β^a needs to be provided at each hop. As far as delay and packet error rate are concerned, given a potential next hop j, on link e_{ij} we can allow a delay δ_{ij} proportional to the geographical advance of the packet towards the destination at that hop. A similar concept

holds for the packet error rate. This can be formalized with $\delta_{ij}^a = \left(\frac{<d_{ij}>_{iN}}{d_{iN}} \right) \cdot \delta^a$ and $\zeta_{ij}^a \leq 1 - \left(1 - \zeta^a \right)^{\lceil \hat{N}_{ij}^{Hop} \rceil^{-1}}$. Here, $< d_{ij} >_{iN}$ (which we refer to as *advance*) is the projection of d_{ij} onto the line connecting node i to the destination, while d_{iN} represents the distance between i and the destination. By assuming that the end-to-end paths will consist of $\hat{N}_{ij}^{Hop}$ hops, we derive the minimum requirement ζ_{ij}^a for the packet error rate for link (i, j).

Admission of flows is regulated by an *admission control protocol*, which works as follows. To establish a contract, each node i broadcasts a short CONTRACT_REQUEST packet, which describes the characteristics of the service required for the set of flows incoming or generated at i, i.e., $\mathcal{F}_i$. If a neighbor j of i i) has positive advance towards the sink N with respect to i, i.e., $j \in \mathcal{P}_i^N$; ii) is able to provide the requested service with the required QoS, i.e., $\beta^a, \delta_{ij}^a, \zeta_{ij}^a, \forall a \in \mathcal{F}_i$, it replies with an ADM_GRANTED control packet. Hence, node i receives an ADM_GRANTED packet from all neighbors able to provide the service. Among these, the optimal relay node j^* is selected according to an optimization criterion described in the following. Node i will then send a CONTRACT_REQUEST packet to the selected node, which will reply with a CONTRACT_ESTABLISHED message that creates the connection. Iteratively, the end-to-end path will be established until the sink is reached. If no ADM_GRANTED message is received, the procedure is aborted and a CONTRACT_RESCINDED message is sent to the upstream node, which will blacklist the downstream node and run the admission control procedure again.

Formally, a local optimization problem is distributively solved by the devices involved, the solution of which determines the optimal data path. Let us introduce the following:

- $E^{pulse} = 2 \cdot E_{elec}^{pulse} + P^{TX} \cdot T_{f,i}$ [J/pulse] accounts for the energy to transmit one pulse from node i to node j, where E_{elec}^{pulse} is the energy per pulse needed by transmitter electronics and digital processing; P^{TX} [W] and $T_{f,i}$ [s] are the average transmitted power and the frame length, respectively.

- $\hat{N}_{ij}^{TX,a}$ is the average number of transmissions of a packet from flow a for the packet to be correctly decoded at receiver j. The actual value $\hat{N}_{ij}^{TX,a} = \frac{1}{1 - PER_{ij}^{\mathcal{C},L}}$ depends on the interference perceived at the receiver, on the coding scheme $\mathcal{C}$ adopted, and on the packet size L.

- $\hat{N}_{ij}^{Hop} = \max \left(\frac{d_{iN}}{<d_{ij}>_{iN}}, 1 \right)$ is the estimated number of hops from i to the destination N when j is next hop.

- $\mathcal{S}_i$ is the *neighbor set* of node i, while $\mathcal{P}_i^N$ is the *positive advance set*, of i, i.e., $j \in \mathcal{P}_i^N$ iff $d_{jN} < d_{iN}$.

- $\mathcal{F}_i$ is the set of incoming or generated flows at node i.

- The bandwidth requirement β^a of application a can be expressed as $\beta^a = R_{0,i}^a \cdot R_{E,i}^a$, where $R_{0,i}^a$ [pulses/s] represents the raw pulse rate for application a required to achieve the rate β^a, when a coding rate $R_{E,i}^a$ is used.

- $\beta^{tot} = \sum_{a \in \mathcal{F}_i} \beta^a$ represents the total bandwidth requirement, in bits/s, for flows incoming or generated at i.

The problem can be cast as follows.

P^{dist}: Distributed Admission Control, Routing and Channel Coding Problem

$$Given: \qquad i, N, \mathcal{S}_i, \mathcal{P}_i^N, E^{pulse}, \mathcal{F}_i$$

$$Find: \qquad j^* \in \mathcal{S}_i \cap \mathcal{P}_i^N, R_{E,i}^a \, \forall a \in \mathcal{F}_i$$

$$Minimize: \quad E_{(i,j)}^{bit} = \frac{1}{\beta^{tot}} \sum_{a \in \mathcal{F}(i)} \frac{E^{pulse} \cdot \hat{N}_{ij}^{TX,a} \cdot \hat{N}_{ij}^{Hop} \cdot \beta^a}{R_{E,j}^a} \quad (1)$$

$$Subject \ to:$$

Packet Error Rate:

$$R_{E,i}^a \le \min \Big(\frac{E_i^{(r)}}{\gamma_{\mathcal{C},j}^a(\zeta_{ij}^a)[\eta_j + \frac{\sigma^2}{T_{f,j}} \sum_{k \in \mathcal{F}(i), k \neq i} E_k^{(r)}]}, 1 \Big), \forall a \in \mathcal{F}_i; \quad (2)$$

Rate Admission Control:

$$\sum_{a \in \mathcal{F}_j} \frac{\beta^a}{R_{E,\mathcal{N}_j}^a(\gamma_{\mathcal{C},\mathcal{N}_j}^a(\zeta_{j\mathcal{N}_j}^a))} +$$

$$+ \sum_{a \in \mathcal{F}_j} \frac{\beta^a}{R_{E,\mathcal{U}_j^a}^a(\gamma_{\mathcal{C},j}^a(\zeta_{\mathcal{U}_j^a j}^a))} + \frac{R_j^{sched,up}}{R_{E,j}^{sched}} + \frac{R_j^{sched,down}}{R_{E,j}^{sched}} \le R_{0,j} \quad (3)$$

Delay Admission Control:

$$\sum_{a \in \mathcal{F}_i} \frac{LR_{0,j}}{R_{E,\mathcal{N}_j}^a} + T^{sched,up} + T^{sched,down} +$$

$$+ \sum_{a \in \mathcal{F}_i} L\Big(1 + \frac{b_j^a}{\phi_j^a}\Big) \cdot \frac{1}{R_{0,j} R_E^a} + \frac{L}{R_{0,j} R_E^a} \le \delta_{ij}^a, \ \forall a \in \mathcal{F}_i. \quad (4)$$

According to the proposed routing rule, i will select j^* as its best next hop iff

$$j^* = argmin_{j \in \mathcal{S}_i \cap \mathcal{P}_i^N} E_{(i,j)}^{bit}, \quad (5)$$

where $E_{(i,j)}^{bit}$ represents the minimum average energy required to successfully transmit a payload bit from node i to the destination, given the interference at j, when i selects j as next hop. This link metric, objective function (1) in **P^{dist}**, takes into account the average number of packet transmissions $\hat{N}_{ij}^{TX,a}$ associated with link (i,j) and flow a. Moreover, it accounts for the average hop-path length $(\hat{N}_{ij}^{Hop})$ from node i to the destination when j is selected as next hop, by assuming that the following hops will guarantee the same advance towards the destination.

Constraint (2) defines the minimum coding rate $R_{E,i}^a$ required at node i to send a packet towards neighbor j in order to guarantee a minimum signal-to-noise-plus-interference (SINR) ratio $\gamma_{\mathcal{C},j}^a(\zeta_{ij}^a)$ at j, i.e., the minimum SINR needed to guarantee a packet error rate ζ_{ij}^a, given the interference generated by the other UWB signals at j (denominator of the expression), as derived in Section IV. Constraint (3) checks if node j has enough bandwidth to satisfy the request, i.e., if the sum of the raw physical data rates of the incoming flows at j (first term

in the sum) plus the outgoing flows (second term) plus the data rate to transmit control packets to determine schedules in the upstream and downstream directions are lower than the raw physical data rate $R_{0,j}$ at j. Here, $\mathcal{N}_j$ represents the next hop of j while $\mathcal{U}_j^a$ represents the upstream node of j for flow a. Finally, constraint (4) checks if j is able to provide service with the required delay. The bound is derived by assuming a wireless fair service approach [5], and extending it for a multi-rate, multi-hop environment with dynamic channel coding with concurrent UWB transmissions, as discussed in Section IV.

IV. MEDIUM ACCESS CONTROL, SCHEDULING AND RATE ASSIGNMENT

In this section, we discuss how our cross-layer module achieves coordination to share the transmission medium among devices, schedules transmissions of data packets and assigns data rates to different flows based on the application requirements.

A. Rate Assignment

The *Signal to Interference plus noise ratio* at node i ($SINR_i$) for a TH-IR-UWB system can be expressed as [6]

$$SINR_i = \frac{P_i g_{ii}}{R_i[\eta_i + \sigma^2 \sum_{j \in \mathcal{F}(i), j \neq i} P_j g_{ji}]}, \ i = 1, \cdots, N, \quad (6)$$

where P_i [W] represents the transmitted power, g_{ij} represents the path loss, R_i [bit/s] represents the data rate on the ith link, and η_i [V^{2}s] represents the background noise energy plus interference from other non UWB systems. Moreover, $T_{f,i}$ [s] represents the length of the physical layer frame on the ith link, while σ is an a-dimensional parameter that depends on the shape of the monocycle. By defining $P_j^{(r)} = P_j g_{ji}$,

$$SINR_i = \frac{E_i^{(r)}}{T_{f,i} R_i[\eta_i + \frac{\sigma^2}{T_{f,i}} \sum_{j \in \mathcal{F}(i), j \neq i} E_j^{(r)}]}, \quad (7)$$

where $E_j^{rec} = P_j^{rec} T_{f,i}$ represents the received energy per pulse from the jth transmitter.

Now, given the allowed PER ζ_i^a at receiver i, it needs to be

$$SINR_i^a \ge \gamma_{\mathcal{C},i}^a(\zeta_i^a), \quad (8)$$

where $\gamma_{\mathcal{C},i}^a(\zeta_i^a)$ is the SINR threshold that guarantees the packet error rate ζ_i^a required by flow a at node i, given the chosen family of error correcting codes $\mathcal{C}$. After some manipulations, and by considering $R_i^a = R_{E,i}^a R_{0,i}$, (8) can be rewritten as

$$R_{E,i}^a \le \min \Big(\frac{E_i^{(r)}}{\gamma_{\mathcal{C},i}^a(\zeta_i^a)[\eta_i + \frac{\sigma^2}{T_{f,i}} \sum_{j \in \mathcal{F}(i), j \neq i} E_j^{(r)}]}, 1 \Big). \quad (9)$$

Hence, the optimal coding rate for flow a is selected as

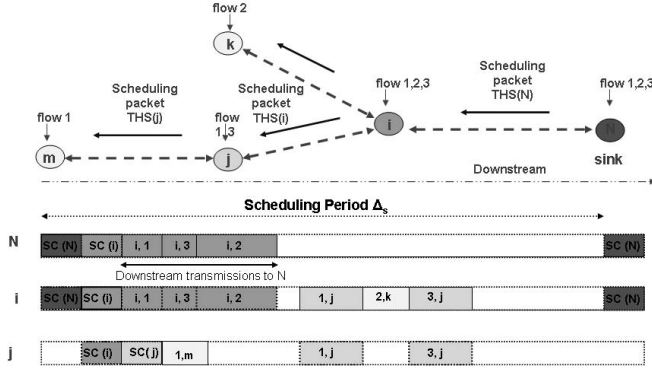$$R_{E,i}^a = \max_{1 \le p \le P} R_E^p \ s.t. \ (9) \ holds \quad (10)$$

Fig. 1. Scheduling of data packets.

B. Receiver-centric Scheduling

For unicast transmissions, a pseudo-random time hopping sequence $THS(j)$ is generated using the identity of the receiver j as the seed of the random number generator, while for multicast transmissions the time hopping sequence $THS(i)$ is generated based on the identity of the transmitter i. Coordination of medium access is still needed to:

1) **Prevent collisions at the receiver.** When a device i is receiving data from a device j, no other device should transmit data intended for i (i.e., using $THS(i)$) simultaneously, as we assume that i is endowed with a simple single-user receiver.

2) **Avoid idle listening.** Each device should be tuned to the wireless channel only when incoming transmissions for itself are occurring, i.e., each device should consume energy only when actually receiving data.

3) **Avoid wasteful transmissions.** When a device i is transmitting data to j, j's receiver must be tuned to $THS(j)$ to listen for incoming transmissions.

Our objective is therefore to devise a medium sharing policy that achieves the above objectives with simple coordination. Our solution is illustrated in Fig. 1. Each device is responsible for scheduling transmissions of data packets from its upstream nodes, i.e., the devices it is offering a service to, i.e., $\forall u \in \mathcal{F}_i$. Device i prepares a SCHEDULE packet, that is transmitted at periodic intervals Δ_s. The scheduling period Δ_s is known to all network devices. The phase Φ_s^i is communicated by i to its upstream nodes in the CONTRACT_ESTABLISHED message. The SCHEDULE packet is broadcast by i and all its upstream nodes receive it by periodically tuning their UWB receiver to $THS(i)$. A *schedule* is a vector of *appointments*, i.e., tuples $(a, u, t_k^a, R_{E,u}^a)$, where a represents an application flow, u represents a node, $u \in \mathcal{F}_i$, t_k^a represents the starting time for transmission of the kth packet from flow a at u, and $R_{E,u}^a$ represents the required coding rate. By sending an appointment $(a, u, t_k^a, R_{E,u}^a)$, node i commits to receiving a packet from u from flow a starting at time t_k^a for a time equal to $L/(R_0 R_E^a)$, where L [bit] is the packet length. Nodes in $\mathcal{F}_i$ transmit a scheduling packet for their upstream nodes, if they have any, immediately after receiving the scheduling

packet from i. Hence, when preparing schedules for their upstream nodes, they can consider previous commitments with their downstream node. In this way, the downstream (closer to sink) node of each node has priority in deciding appointments. Hence, conflict-free scheduling can be achieved in a very simple way. This is only paid in terms of flexibility, as all incoming flows have to be transmitted downstream through the same next-hop, i.e., multi-path routing does not fit in this framework. However, this is a price worth paying for the simplicity achieved. We determine the actual scheduling of packets from upstream nodes based on a procedure inspired by the wireless fair scheduling (WFS) paradigm [5].

Consider a node i, relayer of a set of incoming flows from its upstream nodes $\mathcal{F}_i$. We denote the next hop of i towards the sink N as $\mathcal{N}_i^a$. The kth incoming packet of the ath flow $p_{i,k}^a$ is start-tagged as

$$S(p_{i,k}^a) = \max\{S(p_{i,k-1}^a) + \frac{L_{k-1}^a}{b_i^a}, A(p_{i,k}^a)\}, \qquad (11)$$

where L_{k-1}^a is the length of packet $k-1$ for flow a, b_i^a is called the *bandwidth coefficient*, and $A(p_{i,k}^a)$ represents the arrival time of the packet. The finish tag is set as

$$F(p_{i,k}^a) = S(p_{i,k}^a) + \frac{L_k^a}{\phi_i^a}, \qquad (12)$$

where ϕ_i^a is called the *delay coefficient*. At each step, the scheduler transmits first the packet with the lowest finish time. The bandwidth requirement β^a of flow a can be expressed as $\beta^a = R_{0,i}^a \cdot R_{E,i}^a$. Hence, we define the *bandwidth coefficient*

$b_i^a = \dfrac{R_{0,i}^a}{R_{0,i}^{TOT,IN}} = \dfrac{\frac{\beta^a}{R_{E,i}^a}}{\sum_{b \in \mathcal{F}_i} \frac{\beta^b}{R_{E,i}^b}}$, and the *delay coefficients* $\phi_i^a =$

$1 - \dfrac{\delta_i^a}{\sum_{b \in \mathcal{F}_i} \delta_i^b} = \dfrac{\sum_{b \in \mathcal{F}_i, b \neq a} \delta_i^b}{\sum_{b \in \mathcal{F}_i} \delta_i^b}$.

Note that the value of the bandwidth and delay coefficients, which are fundamental parameters of the schedulers, are constantly updated by the XLCU to reflect the interference perceived at the receiver, changes in paths, and the application requirements so as to assign transmission opportunities that reflect the requirements of the flows being served.

V. PERFORMANCE EVALUATION

We have developed two software simulation tools, i.e., a bit-level physical layer simulator of the TH-IR-UWB communication architecture in Matlab, and a discrete-event object-oriented packet-level simulator in Java. In the simulations presented in this section, the considered packet size is $L = 145$ bytes (125 payload bytes), scheduling packets are sent every $\Delta_s = 10$ ms and the queue size equals to 100 packets.

The considered scenario consists of a 200 m x 200 m terrain where 49 nodes are deployed in a grid structure. There are 2 groups of 12 constant bit rate sources, one located over the lower left corner of the grid, and the other one at the upper left corner. Flows in group 1 require 100 kbit/s bandwidth, 100 ms end-to-end delay, and 0% PER. Flows in group 2 have higher bandwidth demand (500 kbit/s), 100 ms end-to-end delay and can admit 10% PER. The sink is located in the middle of
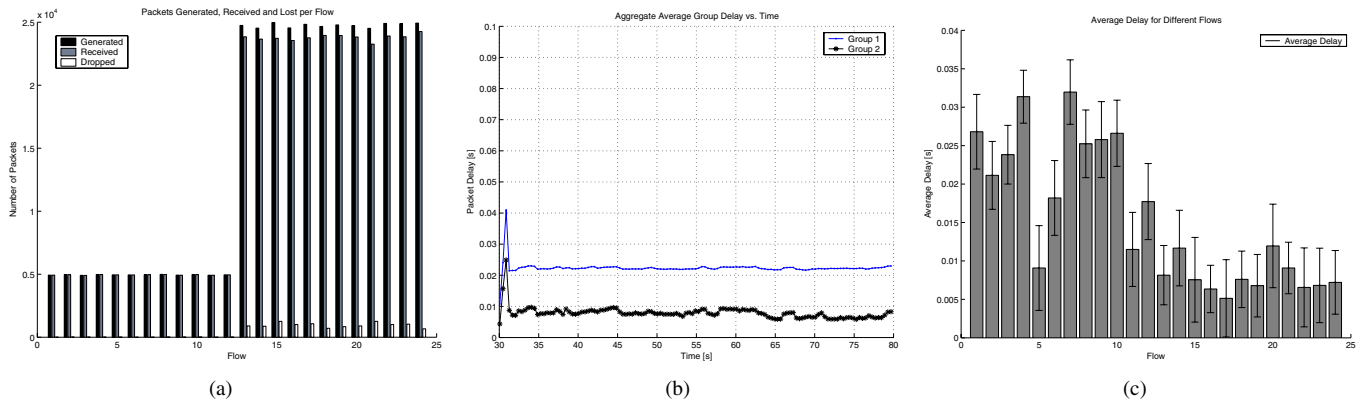
Fig. 3. Scenario 2. Packets Generated, Received and Lost per Flow (a) Aggregate Average Group Delay vs. Time (b) Average Delay for Different Flows (c).
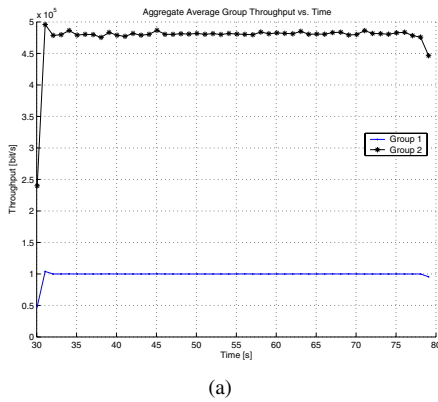

Fig. 2. Aggregate Average Group Throughput vs. Time (c).

the right side of the square. Figure 2(a) shows the average aggregate throughput for sources belonging to the two groups. Sources in group 1 have a throughput of exactly $100\,\mathrm{kbit/s}$, while sources in group 2 show an average throughput of about $480\,\mathrm{kbit/s}$, as some packets are lost. Figure 3(a) shows a bar plot of the packets generated, received and lost per flow. While flows in group 1 do not lose packets, flows in group 2 lose approximately 4% of the packets, which is still below the application requirement. Note that this is achieved with more redundant pulse repetition codes for nodes in group 1. In average, each bit for a flow in group 1 is sent with a coding rate of $1/3$, while the coding rate in group 2 is in average very close to 1. This directly translates into a consistently higher energy consumption. More complex coding schemes can achieve a better energy efficiency at the expense of complexity.

Figure 3(b) shows a comparison between the delays of the two groups with time. The aggregate average end-to-end delays of the two groups are well below the threshold end-to-end delay. The higher delays shown by flows in group 1 are very limited in absolute value (around $10\,\mathrm{ms}$) and are caused by the lower coding rate employed by sources in this group, which lead to higher transmission time. Finally, Fig. 3(c) shows a bar plot of the average end-to-end delay and its variance. The differences in delays between flows in the

same groups are very limited between different flows, which demonstrates the basic fairness of the system, and the variance of the delay is also limited, which shows that under normal circumstances the system leads to much more limited jitter as compared to CSMA/CA based systems.

VI. CONCLUSIONS AND FUTURE WORK

We have described our preliminary design of a cross-layer communication architecture to provide QoS in wireless multimedia sensor networks based on time hopping impulse radio UWB communications. We plan to extend our work in several directions to i) comprehensively evaluate the effect of all system parameters; ii) incorporate adaptive modulation and multi-rate transmission, i.e., to adaptively vary the pulse repetition period (T_f) to trade off multi-user interference for data rate; iii) to provide differentiated service for peak data rate and average data rate; iv) to evaluate and compare alternative scheduling techniques; v) to incorporate in the design adaptive and distributed source coding and end-to-end reliability.

REFERENCES

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdury, "A Survey on Wireless Multimedia Sensor Networks," *Computer Networks (Elsevier)*, vol. 51, no. 4, pp. 921–960, Mar. 2007.

[2] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher, "A Spatiotemporal Communication Protocol for Wireless Sensor Networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. 16, no. 10, pp. 995–1006, October 2005.

[3] E. Felemban, C.-G. Lee, and E. Ekici, "MMSPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks," *IEEE Trans. on Mobile Computing*, vol. 5, no. 6, pp. 738–754, June 2006.

[4] R. Merz, J. Widmer, J.-Y. L. Boudec, and B. Radunovic, "A Joint PHY/MAC Architecture for Low-Radiated Power TH-UWB Wireless Ad-Hoc Networks," *Wireless Communications and Mobile Computing Journal*, vol. 5, no. 5, pp. 567–580, July 2005.

[5] S. Lu, T. Nandagopal, and V. Bharghavan, "Design and analysis of an algorithm for fair service in error-prone wireless channels," *ACM Wireless Networks Journal*, vol. 6, no. 4, January 2000.

[6] F. Cuomo, C. Martello, A. Baiocchi, and F. Capriotti, "Radio Resource Sharing for Ad-hoc Networking with UWB," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 9, pp. 1722–1732, December 2002.