Cross-Layer QoS-Aware Communication for Ultra Wide Band Wireless Multimedia Sensor Networks

Tommaso Melodia, *Member, IEEE,* and Ian F. Akyildiz, *Fellow, IEEE*

***Abstract*—Wireless Multimedia Sensor Networks (WMSNs) are distributed systems of wirelessly networked devices that allow retrieving video and audio streams, still images, and scalar sensor data. WMSNs will be a crucial component of mission-critical networks to protect the operation of strategic national infrastructure, provide support to counteract emergencies and threats, and enhance infrastructure for tactical military operations. To enable these applications, WMSNs 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, whose objective is 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 the modularity of the overall design.**

***Index Terms*—Wireless multimedia sensor networks, cross-layer optimization, quality of service, ultra wide band.**

I. Introduction

WIRELESS Sensor Networks (WSN) have drawn the attention of the research community, driven by a wealth of theoretical and practical challenges. Significant results in this area have ushered in a surge of civil and military applications. As of today, most deployed wireless sensor networks measure scalar physical phenomena like temperature, pressure, humidity, or location of objects. In general, the applications they are designed for have low bandwidth demands, and are usually delay tolerant.

More recently, 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) [2], i.e., networks of wirelessly interconnected devices that can retrieve video and audio streams, still images, and scalar sensor data. By enabling new applications such as multimedia surveillance, traffic enforcement and control systems, advanced health care delivery, structural health monitoring, and industrial process control, WMSNs will be a crucial component of mission-critical networks to protect the operation of strategic national infrastructure, provide support for emergency and crisis intervention, and enhance infrastructure for tactical military operations.

Many of the applications described above require the sensor network paradigm to be re-thought in view of the need to deliver multimedia content with predefined levels of quality of service (QoS). QoS-compliant delivery of multimedia content in sensor networks is a challenging, and largely unexplored task. First, embedded sensors are constrained in terms of battery, memory, processing capability, and achievable data rate [2], while delivery of multimedia flows may be a resource-intensive task. Secondly, in multi-hop wireless networks the attainable capacity of each wireless link depends on the interference level perceived at the receiver. Hence, capacity and delay attainable at each link are location dependent, vary continuously, and may be bursty in nature, thus making QoS provisioning a challenging task. Lastly, functionalities handled at different layers of the communication stack are inherently and strictly coupled due to the shared nature of the communication channel [3]. Hence, different functionalities aimed at QoS provisioning should not be treated separately when efficient solutions are sought, i.e., a cross-layer design approach is needed.

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 considerable energy consumption for idle listening [4]; still, frequent collisions occur due for example to the well-studied *hidden node* problem, in turn leading to increased energy consumption and delays. The transmitted power of currently-off-the-shelf motes, such as Crossbow's MicaZ [5], based on the Chipcon 2420 chipset, is still high, in the order of 1 mW [6]. Exact Tx:Rx:Idle power ratios depend on hardware but idle power is in general not negligible and accounts for a considerable portion of the overall energy consumption.

Manuscript received 1 April 2009; revised 16 November 2009. A preliminary shorter version of this paper [1] was presented in IEEE INFOCOM 2008, Phoenix, AZ. This material is based upon work supported by the National Science Foundation under Grant No. ECCS-0701559.

Tommaso Melodia is with the Wireless Networks and Embedded Systems Laboratory, Department of Electrical Engineering, University at Buffalo, The State University of New York, 332 Bonner Hall, Buffalo, NY 14260 (e-mail: tmelodia@eng.buffalo.edu).

Ian F. Akyildiz is with the Broadband Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA.

Digital Object Identifier 10.1109/JSAC.2010.100604.

Introducing sleep periods to reduce idle listening reduces the energy consumption at the expense of latency and coordination complexity.

For the reasons above, although recent proposals [7][8] have modified existing protocols based on CSMA/CA and geographical routing to provide delay-sensitive and error-resilient services in sensor networks, the application requirements of WMSNs call for a new design perspective and next-generation wireless technologies. Hence, in this paper we propose 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 Ultra Wide Band (UWB) technology has the potential to enable low power consumption, high data rate communications within tens of meters, which make it an ideal choice for WMSNs.

There exist several variants of UWB. The first, known as Time-Hopping Impulse Radio UWB (TH-IR-UWB) [9], is based on sending very short duration pulses (in the order of hundreds of picoseconds) to convey information. As thoroughly discussed in [2], TH-IR-UWB is particularly appealing for WMSNs as it enables high data rate, very low power wireless communications, on simple-design, low-cost radios [10]. Its fine delay resolution properties are well-suited for dense multipath environment [10]. Importantly, interference mitigation techniques [11] allow realizing MAC protocols that do not require mutual temporal exclusion between different transmitters. Finally, the large instantaneous bandwidth enables fine time resolution for accurate position estimation [12] and network synchronization, while at the same time the low power spectral density enables military covert operations.

The remainder of the paper is organized as follows. In Section II, we discuss previous work on multi-hop networking with UWB. In Section III we outline the main design principles, and describe the proposed cross-layer architecture. In Section IV, we introduce the considered system model. In Section V, we describe the routing and admission control functionalities. In Section VI we describe the medium access control and the proposed dynamic code assignment and scheduling policies. In Section VII we discuss performance evaluation results while in Section VIII we draw the main conclusions.

II. Related Work

There is a vast literature on physical layer aspects of the UWB technology. Excellent comprehensive surveys of the UWB transmission technique, and of localization techniques for UWB systems, are provided in [13] and [12], respectively. Although, like CDMA, TH-IR-UWB is a multi-user radio technology, non-zero cross-correlation between time-hopping sequences, time-asynchronicity between sources and the strong effect of multipath propagation require for suitable MAC and higher layer solutions. However, higher layer solutions for multi-hop wireless networking with UWB are not mature yet.

In [14], Cuomo et al. investigate the problem of joint rate and power assignment for TH-IR-UWB, and formulate it as an optimization problem. They show that when the objective is to maximize the aggregate data rate, the optimal solution always corresponds to points where individual devices transmit at the maximum power, or do not transmit at all. The finding is confirmed in [15], where the authors show that power control is not required and may even be suboptimal for wireless networks in the linear regime, i.e., when the achievable data rate is linearly dependent on the *signal-to-interference-plus-noise ratio* (SINR) at the receiver. Note that this is a peculiar characteristic of TH-IR-UWB, and does not hold in general for relatively narrowband systems such as CDMA or IEEE 802.11. The result holds both when the objective is to maximize the data rate under power constraints and when the objective is to minimize the power consumption under constraints on minimum data rates.

Based on the above finding, an uncoordinated MAC protocol for low-power UWB devices is proposed in [11]. While most existing protocols manage interference and multiple-access through power control or mutual exclusion, the MAC proposed in [11] is based on rate control, i.e., it dynamically adapts the channel code based on the interference at the receiver. The proposed design takes advantage of the nature of pulsed TH-UWB to further propose an interference mitigation scheme that alleviates the need for an exclusion scheme. Each device is always allowed to transmit and continuously adapts its channel code to the interference experienced at the destination. Such MAC layer does not need coordination among neighbors that are not involved in the communication, and is shown by simulation to achieve a significant increase in network throughput compared to alternative designs. In [16], a centralized MAC protocol designed to provide QoS support for multimedia traffic in UWB-based wireless local area networks is proposed. However, the protocol is centralized and deals with single-hop networks. In [17], an adaptive medium access control protocol for UWB is proposed, in which nodes periodically declare their current state, so that neighbors can proactively assign power and rate values for new links locally in order to optimize global network performance. In [18], resource management schemes are proposed for QoS support in infrastructure-based UWB networks. In [19], two MAC packet scheduling schemes are proposed, whose objective is to find a suitable tradeoff between system efficiency and fairness. The rate achieved on each link is approximately proportional to its channel quality.

The problem of joint optimal power control, scheduling and routing in UWB networks is dealt with in [20], with the objective of maximizing the aggregate achievable data rate. The problem is formulated as an optimization problem, and is solved approximately for small topologies (up to 50 nodes). Although the paper does not propose practical solutions, it points out important design principles for UWB networks. It is shown that it is optimal to have an exclusion region around the destination, in which all nodes remain silent during transmission, whereas nodes outside of this region can transmit in parallel, regardless of the interference they produce at the destination. As for the routing, it shows that relaying along a minimum energy and loss route is always better than using longer hops or sending directly even if the objective is to maximize the data rate. A similar problem is discussed in

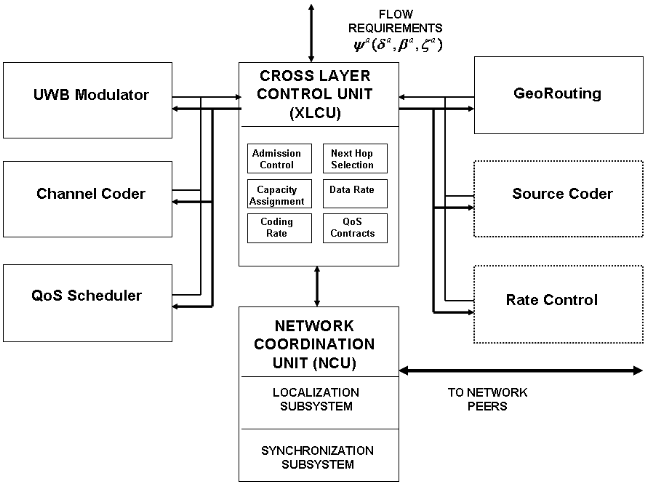


Fig. 1. Architecture of the cross-layer controller.

[21] for sensor networks. A non-linear optimization problem is formulated to to assess the feasibility of relaying data from a set of sensors to a base station. The emphasis is on developing an efficient algorithm to solve the problem. In [22], the authors extend their work to derive the sensor-to-sink capacity of a multi-hop sensor network. The physical layer model considered in [21] and [22] describes accurately a multi carrier UWB system based on CDMA. Hence, the achievable rate is not a linear function of the SINR and most results derived in [20] do not hold. Finally, a vast amount of recent literature has discussed the theoretical foundations of cross-layer design and optimization, which has led to the interpretation of layered protocol design as the result of the application of dual decomposition techniques to nonlinear (often convex) optimization problems. The reader is referred to [23], [24], and [25] for excellent surveys of these approaches.

Unlike our work, none of the previously proposed solutions consider the problem of satisfying and differentiating between QoS requirements of the overlying applications. Moreover, no existing practical solution considers the cross-layer interactions between routing, MAC and physical layer functionalities.

III. Design Principles and Cross-layer Controller

In this section, we overview the principles that guide our system design. We assess the benefits of our design in view of the performance objectives and of the characteristics of WMSNs, and describe the cross-layer control architecture of the UWB sensor.

- **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 architecture of the proposed controller is shown in Fig. 1. 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. In this way, cross-layer interactions can be leveraged without sacrificing on upgradeability, modularity, and ease of system design.
- **Geographical Forwarding.** Time-based localization techniques in UWB allow ranging accuracy in the order of centimeters [12]. Hence, our module leverages geographical information to provide QoS, as further explained in Section V. Positioning capabilities are needed in sensor networks to associate physical meaning to the information gathered by sensors. Moreover, knowledge of the position of each network device allows for scalable routing solutions [26].
- **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 V.
- **Receiver-centric scheduling for QoS Traffic.** In multihop wireless environments interference is location-dependent. For this reason, we provide QoS through receiver-centric scheduling. The receiver can be responsive to the dynamics of the channel 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. Conversely, typical MAC protocols for sensor networks, such as those based on CSMA/CA, require *mutual temporal exclusion* between neighboring transmitters. This allows devising MAC protocols with minimal coordination, as will be discussed in Section VI. In spite of the recent advances in the design of low-complexity transmitters and receivers, the hardware complexity of CDMA transceivers is still relatively high. Instead, TH-IR-UWB transceivers are simple to realize.
- **Dynamic Channel Coding.** As previously discussed, power control is not beneficial in TH-IR-UWB. 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 [11]. This will be explained in Section VI.

IV. System Model

A. Network Model

The sensor network is represented as a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, ..., v_N\}$ is a finite set of devices (nodes) in a finite-dimension terrain, with $N = |\mathcal{V}|$, and $\mathcal{E}$ is the set of links among nodes, i.e., $e_{ij} \in \mathcal{E}$ iff nodes v_i and v_j are within each other's transmission range. Node v_N (also N for simplicity) represents the sink. Each link e_{ij} is also associated with its path loss g_{ij}, as further described in Section IV-C, which is dependent on the distance d_{ij} between nodes v_i and v_j (also i and j for simplicity in the following).

TABLE I
PHYSICAL LAYER AND CHANNEL MODEL

Parameter	Value	Parameter	Value
T_f [s]	$5 \cdot 10^{-8}$	$\Lambda\,[\mathrm{s}^{-1}]$	$0.016 \cdot 10^9$
T_c [s]	$0.18 \cdot 10^{-9}$	$\lambda\,[\mathrm{s}^{-1}]$	$0.22 \cdot 10^9$
τ_p [s]	$5.754 \cdot 10^{-11}$	L	5
δ [s]	$0.03 \cdot 10^{-9}$	K	3
N_h	277	α	1.79

B. Physical Layer Model

TH-IR-UWB transmits subnanosecond pulses (in the order of hundreds of picoseconds), referred to as *monocycles*. We model a monocycle as the second derivative of a Gaussian pulse[1]. Time is slotted in chips of duration T_c, and chips are organized in frames of duration $T_f = N_h T_c$, where N_h is the number of chips per frame. Each user transmits one pulse in one chip per frame, and determines in which chip to transmit based on a pseudo-random *time hopping sequence* (THS). The train of monocycles is modulated based on pulse position modulation (PPM), i.e., a '1' symbol is carried by a monocycle delayed by a time δ with respect to the beginning of the chip, while a '0' symbol begins with the chip. In the above model, the signal $s^{(k)}(t,i)$ generated by the k^{th} user to convey the i^{th} symbol is expressed as

$$s^{(k)}(t,i) = \sqrt{\frac{E_b}{N_0}} \sum_{j=iN_s}^{(i+1)N_s-1} p(t - c_j^{(k)} T_c - jT_f - d_i^{(k)} \delta), \quad (1)$$

where $p(t) = [1 - 4\pi(\frac{t}{\tau_p})^2] \exp(-2\pi(\frac{t}{\tau_p})^2)$ is the second derivative of a Gaussian Pulse, $\{c_j^{(k)}\}$ is the time hopping sequence of the k^{th} source, with $0 \leq c_j^{(k)} \leq N_h - 1$, $\{d_i^{(k)}\}$ is the information-bearing sequence, $d_i^{(k)} \in 0, 1$, E_b represents the energy per bit and N_s the number of pulses to represent a single bit. Clearly, by increasing the number of pulses per bit N_s one can increase the robustness to multiuser interference, at the expense of the data rate, which is expressed as $R = 1/N_s T_f$. This technique is referred to as *pulse repetition coding*. Each transmitter i transmits at a specified *raw pulse rate* $R_{0,i} = 1/T_{f,i}$.

Assuming that pulses generated at the physical layer have a width T_p, we transmit at a peak power $P_{peak} = E_{peak}/T_p = 0.28\,\mathrm{mW}$, i.e., the limits allowed by regulations and hardware constraints [11]. Given a frame of $277\,\mathrm{chips}$, this corresponds to an average radiated power of about $1\,\mu\mathrm{W}$, considerably lower than what radiated by state-of-the-art motes. Table I shows the physical layer parameters considered in most of our experiments, which correspond to a raw pulse rate of $20\,\mathrm{Mpulse/s}$, which seems to be in line with the requirements of WMSNs.

[1]Gaussian pulses are generally used as they can easily be implemented in hardware.

C. Multi-path Channel Model

We model the channel according to the IEEE 802.15.4a standardization group model [27]. The model, specifically developed for sensor network applications, is based on extensive measurements of UWB channels and can be parameterized for indoor residential, indoor office, outdoor, and industrial environments amongst others. The path loss is expressed as

$$g_{ij}\,|_{dB}\,(d_{ij}) = g_0 - 10\alpha \log_{10}(\frac{d_{ij}}{d_0}) \quad (2)$$

where the reference distance d_0 is set to $1\,\mathrm{m}$ and $g_0 = -43.9$. The path loss exponent α depends on whether there is line of sight between the transmitter and the receiver or not, on the antenna gain and efficiency. Note that shadowing can be neglected in 802.15.4a simulations.

The impulse response of the channel is modeled according to a modified Saleh-Valenzuela model [27]. The model reproduces the *clustering phenomenon* observed in several UWB measurements, and accordingly assumes that multipath components arrive in clusters, and that there is independent fading for each cluster and for each ray within the cluster. The impulse response of the channel is given by $h(t) = \sum_{l=0}^{L} \sum_{m=0}^{M} \alpha_{m,l} u_o(t - T_l - \tau_{m,l})$, where $\alpha_{m,l}$ is the multipath gain coefficient for the m^{th} ray in the l^{th} cluster. The interarrival times between two consecutive rays in a cluster and between two consecutive clusters are negative exponentially distributed with parameters λ and Λ, respectively. Hence, the cluster arrival times follow a distribution $P(T_l|T_{l-1}) = \Lambda \exp(-\Lambda(T_l - T_{l-1}))$, $l > 0$, while ray arrival times follow a Poisson distribution of parameter λ. The number of clusters L is Poisson-distributed. The power delay profile (PDP), i.e., the mean power of the different paths, is exponential within each cluster. Table I reports the parameters used for the channel model.

D. Coding

The proposed system includes a *channel encoder* block that encodes raw data bits into encoded bits that are then transmitted as pulses by the UWB modulator. The channel encoder adds redundancy to combat channel impairments and multi-user interference. As discussed in more detail later, our proposed system leverages dynamic channel coding to adapt the transmission rate to the interference perceived at the receiver.

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 $\underline{R}_E = [R_E^1, R_E^2, \cdots, R_E^P]$, where P is the number of different coding rates available and with $R_{E,i}^1 = 1$ (i.e., transmitting uncoded data), and with $R_E^1 > R_E^2 > \cdots > R_E^P$. Hence, when code R_E^p is selected, i.e., when $R_{E,i} = R_E^p$, the encoder produces a block of coded bits of length L/R_E^p. The set of available codes $\underline{R}_E$ depends on the chosen family of codes $\mathcal{C}$. Different families of codes, such as pulse repetition codes or rate-compatible punctured codes, have different performance and different levels of complexity.

E. Traffic Classes

The requirements of an application A are described as a set of tuples $\Psi^A = \{\psi^a(\delta^a, \beta^a, \zeta^a),\ a \in 1, \ldots, N_\psi^A\}$. Here, ψ^a, $a \in 1, \ldots, N_\psi^A$ 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. For example, this allows devising *unequal error protection* schemes in our framework. For example, in an MPEG flow packets belonging to different frames can be protected differently based on their relative importance, e.g., intra-coded frames (I) can be assigned stronger error-protection codes than forward predicted frames (F) or bidirectionally encoded frames (B). An application can leverage layered multiple description codes with the goal of adapting to heterogeneous clients. Base and enhancement layer descriptions can be associated to different subflows, and undergo the admission control check separately. In the remainder of the paper we consider application flows at the level of subflows, i.e., a QoS adapter generates flows with characteristics $\psi^a(\delta^a, \beta^a, \zeta^a)$.

V. Distributed Admission Control Functionality

The proposed system is based on the concept of *Hop-by-Hop QoS contracts*. Each device in the end-to-end path is responsible for locally guaranteeing given performance objectives to devices that are obtaining a service from it. The global, end-to-end requirement is thus guaranteed by the joint local interactions of the participating devices.

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. For example, if the first hop towards the destination guarantees an advance that equals one third of the total geographical distance towards the destination, then one third of the total allowed end-to-end delay can be allowed to that hop. A similar procedure is used to derive the allowable packet error rate on a single hop. This can be formalized by considering

$$\delta_{ij}^a = \left(\frac{< d_{ij} >_{iN}}{d_{iN}}\right) \cdot \delta^a, \tag{3}$$

and

$$\zeta_{ij}^a \leq 1 - \left(1 - \zeta^a\right)^{\lceil \hat{N}_{ij}^{Hop} \rceil^{-1}} \tag{4}$$

In (3), $< d_{ij} >_{iN}$ (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. In (4), by assuming that the end-to-end paths will consist of $\hat{N}_{ij}^{Hop}$ hops, and that the packet error rate will be the same at each hop, 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 node i to the destination N when j is selected as 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.

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

$$\begin{aligned}
&Given: && i,\ N,\ \mathcal{S}_i,\ \mathcal{P}_i^N,\ E^{pulse},\ \mathcal{F}_i \\
&Find: && j^* \in \mathcal{S}_i \cap \mathcal{P}_i^N,\ R_{E,i}^a\ \forall a \in \mathcal{F}_i \\
&Minimize: && E_{(i,j)}^{bit} = \frac{1}{\beta^{tot}} \sum_{a \in \mathcal{F}(i)} \frac{E^{pulse} \cdot \hat{N}_{ij}^{TX,a} \cdot \hat{N}_{ij}^{Hop} \cdot \beta^a}{R_{E,j}^a} \qquad (5) \\
&Subject\ to:
\end{aligned}$$

Packet Error Rate: See (6).

Rate Admission Control:

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

$$R^a_{E,i} \le \min\Big(\frac{E_i^{(r)}}{\gamma^a_{\mathcal{C},j}(\zeta^a_{ij})[\eta_j + \frac{\sigma^2}{T_{f,j}}\sum_{k\in\mathcal{F}(i),k\neq i} E_k^{(r)}]}, 1\Big),\ \forall a \in \mathcal{F}_i; \quad (6)$$

$$+\sum_{a\in\mathcal{F}_j}\frac{\beta^a}{R^a_{E,\mathcal{U}^a_j}(\gamma^a_{\mathcal{C},j}(\zeta^a_{\mathcal{U}^a_j j}))} + \frac{R_j^{sched,up}}{R_{E,j}^{sched}} + \frac{R_j^{sched,down}}{R_{E,j}^{sched}} \le R_{0,j} \quad (7)$$

Delay Admission Control:

$$\sum_{a\in\mathcal{F}_i}\frac{LR_{0,j}}{R^a_{E,\mathcal{N}_j}} + T^{sched,up} + T^{sched,down} +$$
$$+\sum_{a\in\mathcal{F}_i} L\Big(1+\frac{b^a_j}{\phi^a_j}\Big)\cdot\frac{1}{R_{0,j}R^a_{E,j}} + \frac{L}{R_{0,j}R^a_{E,j}} \le \delta^a_{ij},\ \forall a \in \mathcal{F}_i. \quad (8)$$

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

$$j^* = argmin_{j\in\mathcal{S}_i\cap\mathcal{P}_i^N} E^{bit}_{(i,j)}, \quad (9)$$

where $E^{bit}_{(i,j)}$ 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 (5) in $\mathbf{P^{dist}}$, takes into account the average number of packet transmissions $\hat{N}^{TX,a}_{ij}$ associated with link (i,j) and flow a. Moreover, it accounts for the average hop-path length $(\hat{N}^{Hop}_{ij})$ 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. While this is a simple way to estimate the number of hops towards the destination, i) it does not incur any signaling overhead; ii) its accuracy increases as the density increases; iii) its accuracy increases as the distance to the destination decreases.

Note that this relatively complex problem nicely decomposes and can be solved in a distributed way. The solution can be interpreted as decoupling $\mathbf{P^{dist}}$ into three sub-problems: first, at each feasible next hop (neighbor with positive advance), find, if it exists, the minimum-redundancy coding rate for i to meet the local PER requirement ζ^a_{ij} for each flow a in $\mathcal{F}_i$ (constraint (6)); note that this is straightforward since it implies finding the maximum (minimum-redundancy) coding rate such that (6)) holds, since the objective function monotonically decreases with increasing coding rates. Second, check if given the required coding rates, node j has sufficient bandwidth (constraint (7)) and can provide service to the flows with the required delay (constraint (8)). In practice, this first three steps are performed at each node receiving the `ADM_REQUEST` packet. Finally, among the nodes that have granted admission, node i picks the node j^* with minimal link metric given the chosen coding rate. Note that the problem solved at each node i has a low computational overhead, i.e., proportional to the number of its neighboring nodes with positive advance that are able to provide the requested service.

Constraint (6) defines the minimum-redundancy coding rate $R^a_{E,i}$ required at node i to send a packet towards neighbor j in order to guarantee a minimum signal-to-noise-plus-interference (SINR) ratio $\gamma^a_{\mathcal{C},j}(\zeta^a_{ij})$ at j, i.e., the minimum SINR needed to guarantee a packet error rate ζ^a_{ij}, given the interference generated by the other UWB signals at j (denominator of the expression), as derived in Section VI. Constraint (7) 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 $\mathcal{U}^a_j$ represents the upstream node of j for flow a. Finally, constraint (19) checks if j is able to provide service with the required delay. The bound is derived by assuming a wireless fair service approach [28], and extending it for a multi-rate, multi-hop environment with dynamic channel coding with concurrent UWB transmissions, as further discussed in Section VI.

VI. 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 [14]

$$SINR_i = \frac{P_i g_{ii}}{R_i[\eta_i + \sigma'^2 T_{f,i}\sum_{j\in\mathcal{F}(i),j\neq i} P_j g_{ji}]}, i = 1,\cdots,N, \quad (10)$$

where P_i [W] represents the transmitted power, g_{ij} represents the path loss, R_i [bit/s] represents the data rate on the i^{th} 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 i^{th} link, while σ' is an a-dimensional parameter that depends on the shape of the monocycle and on the frame length. Note that in this paper we assume that all links have the same rate, i.e., $T_{f,i} = T_f$. However, our solution can be extended to a multi-rate system by considering the expression for the SNR in a multi-rate TH-IR-UWB system as derived in [29]. We can express σ'^2 as $\sigma'^2 = \frac{1}{T_f}\cdot\frac{\int_{-\infty}^{+\infty}\left[\int_{-\infty}^{+\infty}p(x-\tau)v(x)dx\right]^2 d\tau}{\int_{-\infty}^{+\infty}p(x-\delta)v(x)dx} = \frac{1}{T_f}\sigma^2$, where p represents the received impulse shape and $v(t) = p(t) - p(t-\delta)$ represents the *correlator's template signal* at the receiver. By defining $P_j^{(r)} = P_j g_{ji}$,

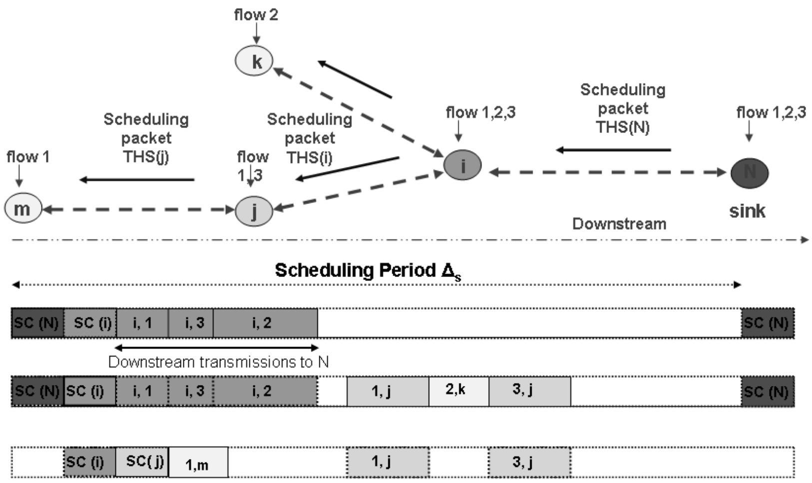


Fig. 2. Scheduling of data packets.

$$SINR_i = \frac{E_i^{(r)}}{T_{f,i} R_i [\eta_i + \frac{\sigma^2}{T_{f,i}} \sum_{j \in \mathcal{F}(i), j \neq i} E_j^{(r)}]}, \tag{11}$$

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

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

$$SINR_i^a \geq \gamma_{\mathcal{C},i}^a(\zeta_i^a), \tag{12}$$

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 $R_i^a = R_{E,i}^a R_{0,i}$, (12) can be rewritten as

$$R_{E,i}^a \leq \min\Big(\frac{E_i^{(r)}}{\gamma_{\mathcal{C},i}^a(\zeta_i^a)[\eta_i + \frac{\sigma^2}{T_{f,i}} \sum_{j \in \mathcal{F}(i), j \neq i} E_j^{(r)}]}, 1\Big). \tag{13}$$

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

$$R_{E,i}^a = \max_{1 \leq p \leq P} R_E^p \, s.t. \, (13) \, holds \tag{14}$$

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. 2. 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 k^{th} 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 $L\,[\text{bit}]$ 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. WFS [30] is a family of solutions designed to guarantee delay-bounded and throughput-guaranteed access in single-hop, single-rate wireless packet networks (i.e., cellular networks). Most of these solutions are based on wireless adaptations of the packetized version of the Generalized Processor Sharing (GPS) paradigm [31]. We consider a wireless fair service approach [28] and extend it for a UWB multi-rate, multi-hop environment.

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 k^{th} incoming packet of the a^{th} flow $p_{i,k}^a$ is start-tagged as

$$S(p_{i,k}^a) = \max\{S(p_{i,k-1}^a) + \frac{L_{k-1}^a}{b_i^a}, A(p_{i,k}^a)\}, \quad (15)$$

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

$$F(p_{i,k}^a) = S(p_{i,k}^a) + \frac{L_k^a}{\phi_i^a}, \quad (16)$$

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 as

$$b_i^a = \frac{R_{0,i}^a}{R_{0,i}^{TOT,IN}} = \frac{\frac{\beta_i^a}{R_{E,i}^a}}{\sum_{b\in\mathcal{F}_i}\frac{\beta_i^b}{R_{E,i}^b}}. \quad (17)$$

We define the *delay coefficients* ϕ_i^a as

$$\phi_i^a = 1 - \frac{\delta_i^a}{\sum_{b\in\mathcal{F}_i}\delta_i^b} = \frac{\sum_{b\in\mathcal{F}_i,\, b\neq a}\delta_i^b}{\sum_{b\in\mathcal{F}_i}\delta_i^b}. \quad (18)$$

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. With the above definitions, the new queue delay $D_j^{a,new}$ of the head of line packet of flow a is bounded by

$$D_j^{a,new} \leq \sum_{a\in\mathcal{F}_i}\frac{LR_{0,j}}{R_{E,\mathcal{N}_j}^a} + T^{sched,up} + T^{sched,down} +$$
$$+ \sum_{a\in\mathcal{F}_i} L\Big(1 + \frac{b_j^a}{\phi_j^a}\Big)\cdot\frac{1}{R_{0,j}R_{E,j}^a} + \frac{L}{R_{0,j}R_{E,j}^a}. \quad (19)$$

The expression above follows by extending theorem 4.3 in [28] to the case when the scheduler is relaying data to an upstream node. The above bound, where $T^{sched,up}$ and $T^{sched,down}$ represent the time needed to transmit the schedule packets upstream and downstream, respectively, is used by the admission control procedure, as explained in Section V.

VII. Performance Evaluation

To assess the performance of the proposed solution, 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. The physical layer simulator models generation, modulation and coding of Gaussian monocycles, convolution with the multi-path-affected UWB channel, interference from concurrent transmitters, and reception with a correlation receiver affected by multi-user interference and noise as described in Section IV-B.

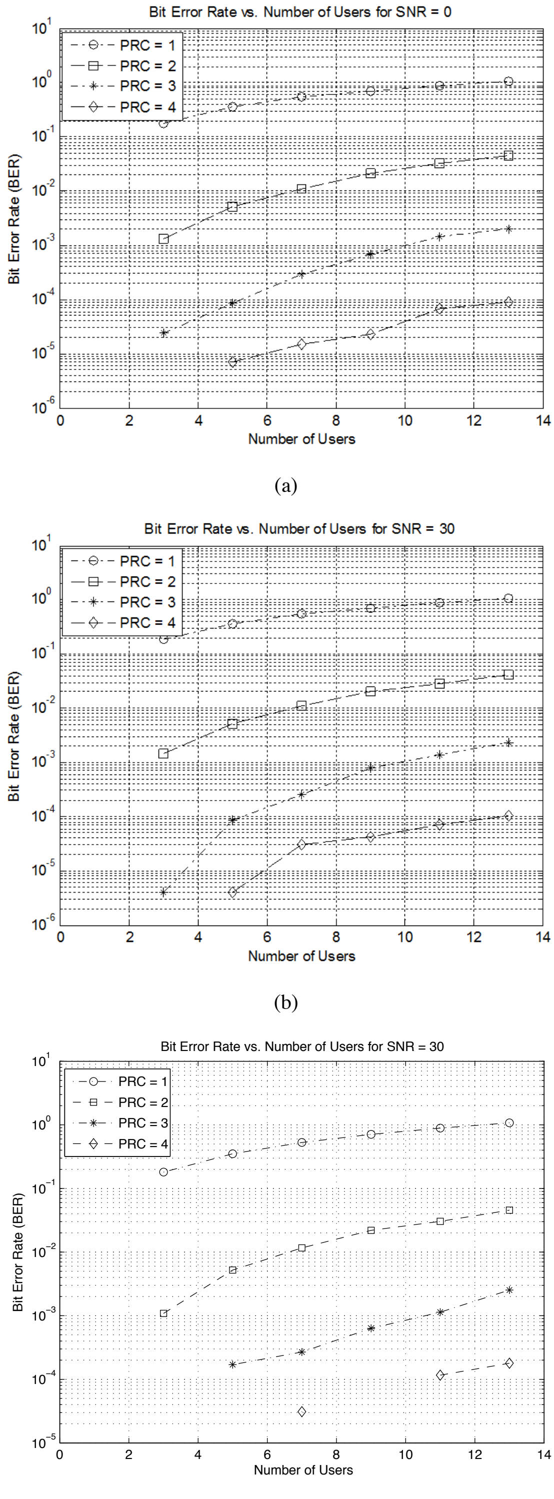
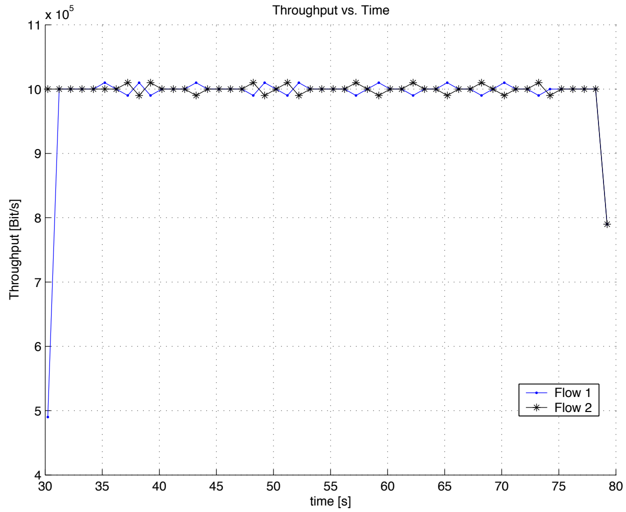
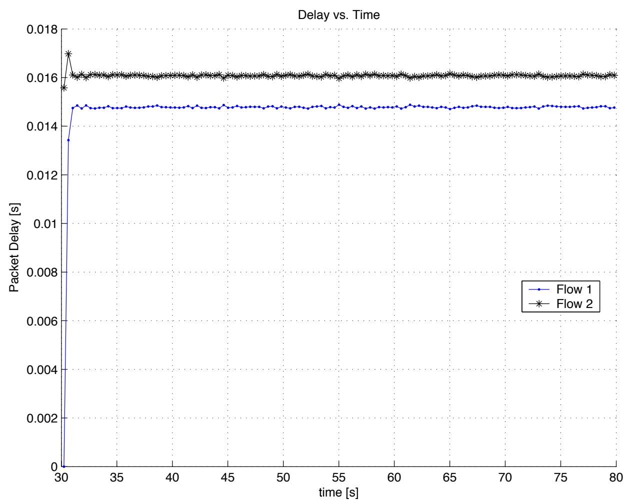


Fig. 3. Bit Error Rate with increasing number of users, for different Pulse Repetition Codes, for SNR=30dB, no multipath (a) for SNR=30dB, no multipath (b) for SNR=30dB, with multipath (c).

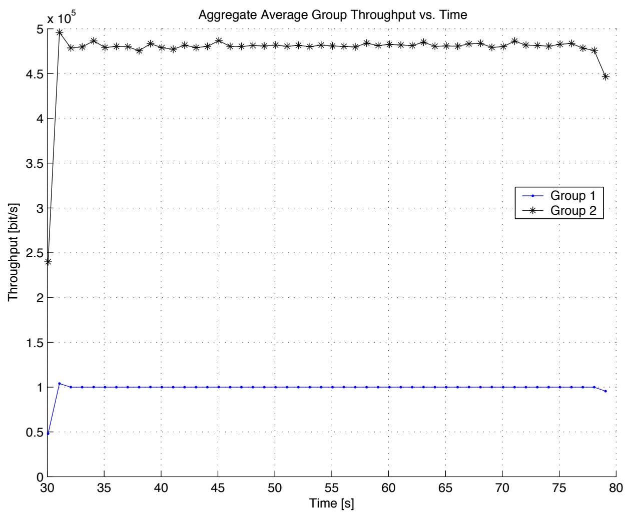
An extensive simulation campaign provided us with results, expressed in terms of bit error rate versus channel characteristics, number of interferers, and SNR, which have



(a)



(b)



(c)

Fig. 4. Scenario 1. Throughput vs. Time (a) and Delay vs. Time (b) for two different flows. Scenario 2. Aggregate Average Group Throughput vs. Time (c).

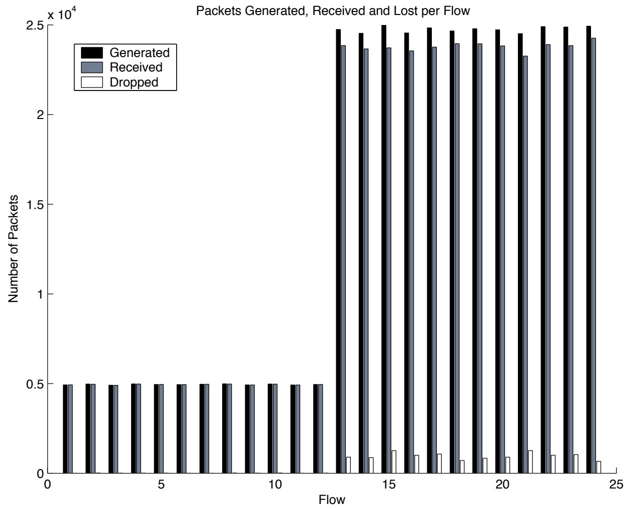
then been plugged into the packet-level simulator developed in Java. For each point in the figure, we have repeated simulations with a block of $1000\,\mathrm{bits}$ until the relative error, i.e., the ratio between the single-sided 95% confidence interval and the estimated value, is below 10%. Figures 3(a) and 3(b), show the Bit Error Rate with increasing number of users (i.e., interference) for different Pulse Repetition Codes, for SNR=0dB and SNR=30dB, respectively, for the case of channel without multipath, while Fig. 3(c) refers to a channel highly affected by multipath (see Table I). While the PRC code used considerably affects the attainable BER, in TH-IR-UWB the SNR does not influence the BER to the same extent. This can be ascribed to the structure of the correlator receiver.

The Java simulator models all aspects of the communication architecture described in this paper. In the simulations presented in this section, the considered packet size is $L = 145\,\mathrm{bytes}$ (125 payload bytes), scheduling packets are sent every $\Delta_s = 10\,\mathrm{ms}$ and the queue size equals to $100\,\mathrm{packets}$.

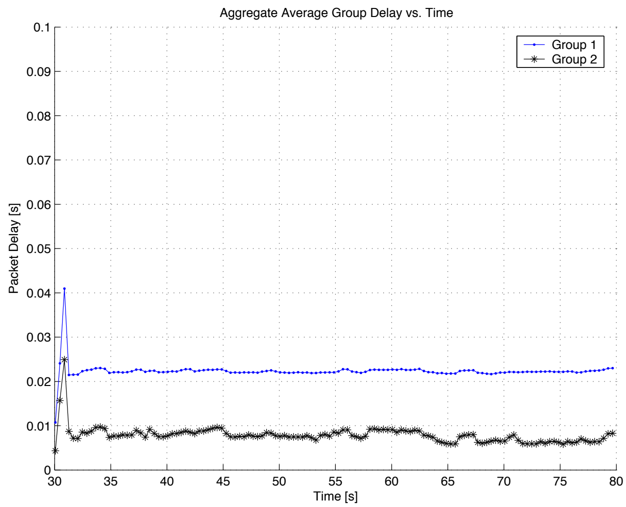
Figures 4(a) and 4(b) are from a simple scenario where there are two sources, both generating traffic requiring 1Mb/s, and with equal loose delay requirements. Figure 4(a) shows that both sources receive in average the required service, and Fig. 4(b) shows that the end-to-end packet delivery delays are very short (around $15\,\mathrm{ms}$), do not fluctuate (low jitter), and are comparable for the two sources. The difference in the delays is caused by unsynchronized packet generation at the sources.

The second considered scenario consists of a $200\,\mathrm{m} \times 200\,\mathrm{m}$ 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 $100\,\mathrm{kbit/s}$ bandwidth, $100\,\mathrm{ms}$ end-to-end delay, and 0% PER. Flows in group 2 have higher bandwidth demand ($500\,\mathrm{kbit/s}$), $100\,\mathrm{ms}$ end-to-end delay and can admit 10% PER. The sink is located in the middle of the right side of the square. Figure 4(c) 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 5(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 considerably higher energy consumption. More complex coding schemes can achieve a better energy efficiency at the expense of complexity.

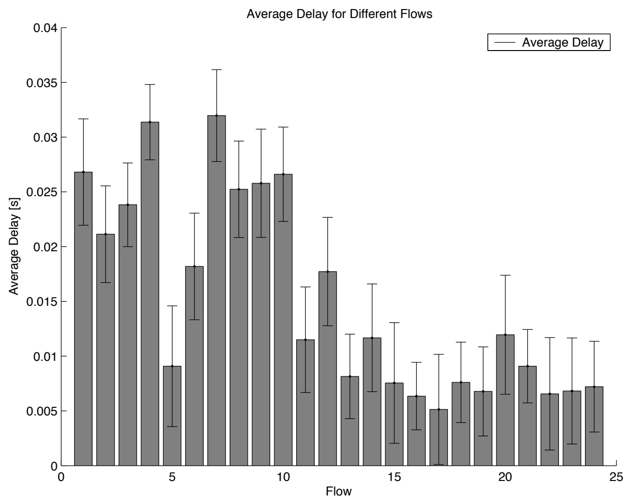
Figure 5(b) shows a comparison between the delays of the two groups with time. The aggregate average end-to-end delays of the two groups are well below the threshold end-to-end delay. The higher delays shown by flows in group 1 are very limited in absolute value (around $10\,\mathrm{ms}$) and are caused by the lower coding rate employed by sources in this group, which lead to higher transmission time. Finally, Fig. 5(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



(a)



(b)



(c)

Fig. 5. Scenario 2. Packets Generated, Received and Lost per Flow (a) Aggregate Average Group Delay vs. Time (b) Average Delay for Different Flows (c).

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.

VIII. Conclusions

We have described the design of a cross-layer communication architecture to provide QoS in wireless multimedia sensor networks based on time hopping impulse radio UWB communications. The architecture is based on an innovative design that aims at providing differentiation in the domains of throughput, delay, reliability, based on a modular cross-layer controller that performs admission control, routing, scheduling, bandwidth assignment and coding to satisfy application requirements. Performance evaluation shows that the architecture is a promising solution to satisfy the performance targets of WMSNs. In particular, delays are very low and with low jitter, and throughput is fairly constant in time.

References

[1] T. Melodia and I. F. Akyildiz, "Cross-layer Quality of Service Support for UWB Wireless Multimedia Sensor Networks," in *Proc. IEEE Conference on Computer Communications (INFOCOM), Mini-Conference*, Phoenix, AZ, Apr. 2008.

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

[3] M. Chiang, "Balancing Transport and Physical Layers in Wireless Multihop Networks: Jointly Optimal Congestion Control and Power Control," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 104 – 116, January 2005.

[4] G. J. Pottie and W. J. Kaiser, "Wireless Integrated Network Sensors," *Communications of the ACM*, vol. 43, pp. 51–58, May 2000.

[5] "Crossbow MICAz Mote Specifications," http://www.xbow.com.

[6] I. Akyildiz, T. Melodia, and K. Chowdhury, "Wireless multimedia sensor networks: Applications and testbeds," *Proc. IEEE*, vol. 96, no. 10, pp. 1588–1605, Oct. 2008.

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

[8] 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. Mobile Comput.*, vol. 5, no. 6, pp. 738–754, June 2006.

[9] J. Reed, *Introduction to Ultra Wideband Communication Systems*. Englewood Cliffs, New Jersey: Prentice Hall, June 2005.

[10] R. A. Scholtz and N. Z. Win, "Impulse Radio: How it Works," *IEEE Commun. Lett.*, vol. 2, no. 2, pp. 36–38, February 1998.

[11] 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.

[12] S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via Ultra-Wideband Radios," *IEEE Signal Processing Mag.*, vol. 22, no. 4, pp. 70–84, July 2005.

[13] L. Yang and G. B. Giannakis, "Ultra-WideBand Communications: An Idea Whose Time Has Come," *IEEE Signal Processing Mag.*, vol. 3, no. 6, pp. 26–54, November 2004.

[14] F. Cuomo, C. Martello, A. Baiocchi, and F. Capriotti, "Radio Resource Sharing for Ad-hoc Networking with UWB," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 9, pp. 1722–1732, December 2002.

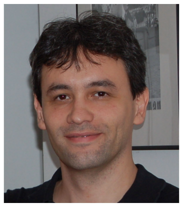
[15] B. Radunovic and J.-Y. L. Boudec, "Power Control is Not Required for Wireless Networks in the Linear Regime," in *Proc. of IEEE Intl. Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Taormina - Giardini Naxos, Italy, June 2005.

[16] Y. Chu and A. Ganz, "A Centralized MAC Protocol for QoS Support in UWB-Based Wireless Networks," *Wirel. Pers. Commun.*, vol. 34, no. 1-2, pp. 45–66, 2005.

[17] R. Jurdak, P. Baldi, and C. V. Lopes, "U-MAC: a proactive and adaptive UWB medium access control protocol," *Wirel. Commun. Mob. Comput.*, vol. 5, no. 5, pp. 551–566, August 2005.

[18] G. Giancola, C. Martello, F. Cuomo, and M.-G. Di Benedetto, "Radio resource management in infrastructure-based and ad hoc UWB networks," *Wirel. Commun. Mob. Comput.*, vol. 5, no. 5, pp. 581–597, 2005.

[19] H. Jiang and W. Zhuang, "Effective Packet Scheduling with Fairness Adaptation in Ultra Wideband Wireless Networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 2, pp. 680–690, February 2007.
[20] B. Radunovic and J.-Y. Le Boudec, "Optimal Power Control, Scheduling and Routing in UWB Networks," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 7, p. 1252, 2004.
[21] Y. Shi, Y. Hou, H. Sherali, and S. Midkiff, "Cross-layer optimization for routing data traffic in UWB-based sensor networks," in *Proc. ACM Intl. Conf. on Mobile Computing and Networking (MobiCom)*, Cologne, Germany, August-September 2005.
[22] Y. Shi and Y. T. Hou, "On the capacity of UWB-based wireless sensor network," *Computer Networks (Elsevier)*, vol. 52, no. 14, pp. 2797–2804, Oct. 2008.
[23] X. Lin, N. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1452–1463, Aug. 2006.
[24] D. P. Palomar and M. Chiang, "A Tutorial on Decomposition Methods for Network Utility Maximization," *IEEE J. Sel. Areas Commun.: Special Issue on Nonlinear Optimization of Communication Systems*, vol. 24, no. 8, pp. 1439–1451, Aug. 2006.
[25] M. Chiang, S. Low, A. Calderbank, and J. Doyle, "Layering as Optimization Decomposition: A Mathematical Theory of Network Architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, January 2007.
[26] T. Melodia, D. Pompili, and I. F. Akyildiz, "On the Interdependence of Distributed Topology Control and Geographical Routing in Ad Hoc and Sensor Networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 3, pp. 520–532, Mar. 2005.
[27] A. F. Molisch, K. Balakrishnan, D. Cassioli, C. Chong, S. Emami, A. Fort, J. Karedal, J. Kunisch, H. Schantz, U. Schuster, and K. Siwiak, "A comprehensive model for ultrawideband propagation channels," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, Saint Louis, MO, November 2005.
[28] 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.
[29] H. Wymeersch, G. Zussman, and M. Win, "SNR analysis for multi-rate UWB-IR," *IEEE Commun. Lett.*, vol. 11, no. 1, pp. 49–51, Jan. 2007.
[30] V. Bharghavan, S. Lu, and T. Nandagopal, "Fair queueing in wireless networks: Issues and approaches," *IEEE Pers. Commun. Mag.*, vol. 6, no. 1, February 1999.
[31] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: The single node case," *IEEE/ACM Trans. Netw.*, vol. 1, no. 3, pp. 344–357, June 1993.



Tommaso Melodia [M'2007] (tmelodia@eng.buffalo.edu) is an Assistant Professor with the Department of Electrical Engineering at the University at Buffalo, The State University of New York (SUNY), where he directs the Wireless Networks and Embedded Systems Laboratory. He received his Ph.D. in Electrical and Computer Engineering from the Georgia Institute of Technology in June 2007. He had previously received his "Laurea" (integrated B.S. and M.S.) and Doctorate degrees in Telecommunications Engineering from the University of Rome ñLa Sapienza,î Rome, Italy, in 2001 and 2005, respectively. He is the recipient of the BWN-Lab Researcher of the Year award for 2004.He coauthored a paper that was was recognized as the *Fast Breaking Paper in the field of Computer Science* for February 2009 by Thomson ISI Essential Science Indicators. He is an Associate Editor for the Computer Networks (Elsevier) Journal, Transactions on Mobile Computing and Applications (ICST) and for the Journal of Sensors (Hindawi). He serves in the technical program committees of several leading conferences in wireless communications and networking, including IEEE Infocom, ACM Mobicom, and ACM Mobihoc. He was the technical co-chair of the Ad Hoc and Sensor Networks Symposium for IEEE ICC 2009. His current research interests are in modeling and optimization of multi-hop wireless networks, cross-layer design and optimization, wireless multimedia sensor and actor networks, underwater acoustic networks, and cognitive radio networks.

Ian F. Akyildiz [F'1995] (ian@ece.gatech.edu) is the Ken Byers Distinguished Chair Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology. Since June 2008, he is an Honorary Professor with the School of Electrical Engineering at the Universitat Politecnico de Catalunya, Barcelona, Spain. He is the Editor-in-Chief of *Computer Networks (COMNET) Journal* as well as the founding Editor-in-Chief of the *Ad Hoc Networks Journal* and *Physical Communication Journal* all with Elsevier. His current research interests are in cognitive radio networks, wireless sensor networks and nano-communication networks.

He received numerous awards, including the 1997 IEEE Leonard G. Abraham Prize Award (IEEE Communications Society) for his paper entitled "Multimedia Group Synchronization Protocols for Integrated Services Architectures" published in the IEEE JSAC in January 1996; the 2002 IEEE Harry M. Goode Memorial Award (IEEE Computer Society) with the citation "for significant and pioneering contributions to advanced architectures and protocols for wireless and satellite networking"; the 2003 IEEE Best Tutorial Award (IEEE Communication Society) for his paper entitled "A Survey on Sensor Networks," published in IEEE Communications magazine, in August 2002; the 2003 ACM Sigmobile Outstanding Contribution Award with the citation "for pioneering contributions in the area of mobility and resource management for wireless communication networksÓ; the 2004 Georgia Tech Faculty Research Author Award for his "outstanding record of publications of papers between 1999-2003Ó; the 2005 Distinguished Faculty Achievement Award from School of ECE, Georgia Tech; the 2009 Georgia Tech Outstanding Doctoral Thesis Advisor Award for his 20+ years service and dedication to Georgia Tech and producing outstanding PhD students; and the 2009 ECE Distinguished Mentor Award from School of ECE, Georgia Tech. He has been a Fellow of the Association for Computing Machinery (ACM) since 1996.