Scheduling Algorithms for Multimedia Traffic Over High–rate WPANs

Andrea Zanella and Fabio Lorquando, *Member*, IEEE

Abstract — *The IEEE 802.15.3 standard aims at covering the gaps of current Wireless Personal Area Network (WPAN) technologies in supporting applications with very high–rate and/or quality of service requirements. To this end, the standard encompasses high–rate modulations and a flexible medium access mechanism that permits resource reservation. Whereas the standard defines the general resource management framework, the definition of practical scheduling algorithms is left open to proprietary solutions. In this paper we investigate the potentialities offered by the IEEE 802.15.3 framework for supporting multimedia services. More specifically, we analyze the performance of some classical scheduling policies in presence of intensive real–time and multimedia traffic, in order to identify the most effective strategy for the considered scenarios. The analysis has been performed by using a complete 802.15.3 C++ simulator, where we have realized the different scheduling strategies upon an entirely standard–compliant round robin polling procedure. Results show that the simple and well-known scheduling strategies considered in this study offer very good performance in a large set of realistic application scenarios.*[1]

Index Terms — Multimedia, high rate WPAN, quality of service, resource management, scheduling.

I. INTRODUCTION

The standardization effort of the IEEE 802.15.3 group is intended to provide High–Rate Wireless Personal Area Network (HR–WPAN) technologies for supporting the always growing demand for mobile connectivity, easy data sharing and inter–operability among electronic devices of different nature. In particular, WPAN technologies are expected to permit wireless fruition of multimedia services, such as video streaming, voice over IP, multi–player gaming, which have been experimenting an impressive diffusion in the last years. The realization of the envisioned scenario requires the definition of suitable radio resource management schemes, able to provide each application with the required Quality of Service (QoS) level [1]. Resource sharing is a classical and well–known problem, which has been addressed in several different contexts, leading to the definition of many scheduling algorithms of different complexity.

In this paper, we address the problem of supporting heterogeneous multimedia flows, such as in MPEG-4 video, Voice over IP, and interactive gaming, in HR-WPANs, from a practical perspective. Instead of searching novel scheduling schemes, whose complexity might overtake the performance benefits, we focus our attention on classical, well–established algorithms, extensively tested in other scenarios, that might be readily ported on this novel networking platform. More specifically, we compare four well–known scheduling algorithms, namely Generalized Processor Sharing (GPS), Earliest Deadline First (EDF), EDF with Discard (EDF–DS) and EDF with Soft/Hard deadlines (EDF–SH) in different application scenarios. To this end, we define a standard–compliant polling mechanism that permits the network coordinator to gather the traffic information from the source devices and schedule the transmission resources as dictated by the considered scheduling algorithm. The rest of the paper is organized as follows. Section II briefly overviews the 802.15.3 standard, with particular attention to the Medium Access Control (MAC) scheme. Section III reviews the literature on the topic. Section IV describes the scheduling algorithms adopted in the work. Section V proposes a standard compliant polling scheme that might be used to gather the traffic information required by the scheduling algorithms to work. The traffic models employed in the study and the simulation settings are described in Section VI, whereas simulation results are presented and discussed in Section VII. Finally, Section VIII concludes the paper with some final considerations.

II. IEEE 802.15.3 BASICS

The 802.15.3 standard defines a Wireless Personal Area Network (WPAN) with a limited spatial extension, able to support high data rates (ranging from 11 up to 55 Mb/s) and QoS oriented.

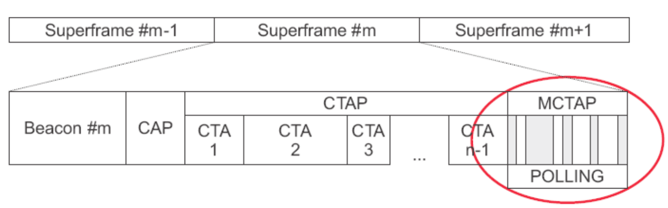

Fig. 1. General superframe structure.

The network topology of IEEE 802.15.3 closely resembles Bluetooth's one: the HR–WPAN is called *piconet* and it is composed by a set of *devices*, denoted by DEV, logically associated to a *piconet coordinator*, indicated by PNC. The PNC controls and manages the piconet functioning through the periodic broadcasting of an informative *beacon* message.

[1]Andrea Zanella is with the Information Engineering Department, University of Padova, Padova, 35131 Italy (e-mail: zanella@dei.unipd.it).
Fabio Lorquando was with the Information Engineering Department, University of Padova, Padova, 35131 Italy. He is currently working at Altran Group as a TLC/media consultant. (e-mail: fabio.lorquando@gmail..com).

Contributed Paper
Manuscript received May 30, 2008

Every beacon begins a time interval, named *superframe*, of variable duration (up to 65535 μs). The beacon is usually followed by a Contention Access Period (CAP) that, in turn, is followed by a Channel Time Allocation Period (CTAP). During CAP, channel access is governed by the usual Carrier–Sense Multiple Access – Collision Avoidance (CSMA/CA) protocol, whereas during CTAP a collision–free Time Division Multiple Access (TDMA) scheme is adopted. The CTAP, in fact, is arranged in uneven time slots, called Channel Time Allocations (CTAs) and Management Channel Time Allocations (MCTAs). The general superframe structure is graphically illustrated in Fig. 1.

CTAs are univocally assigned to a communication flow identified by the triplet <*Stream ID, Source DEV, Destination DEV*>, where *Source DEV* and *Destination DEV* are MAC addresses of the source and destination nodes, respectively, while *Stream ID* is an identifier associated to a specific data stream between the two nodes. CTAs can be either static or dynamic. Once allocated, a static CTA occurs periodically until the owner DEV explicitly dismisses the service. Static CTAs are particularly suitable for isochronous and constant bit rate traffic sources and the repetitiveness of their allocation make it robust against beacon reception errors. Dynamic CTAs, conversely, can be modified on a superframe-by-superframe basis, according to the scheduling policy implemented by the PNC. Therefore, the position and duration of dynamic CTAs are indicated in the beacon that begins each superframe. The flexibility in the resource management permitted by dynamic CTAs is paid in terms of robustness to beacon losses: whenever a DEV fails decoding the beacon message, it cannot make use of any dynamic CTAs in that superframe.

Within a CTA, the source DEV can transmit multiple frames to the destination DEV that, in turn, can send the corresponding acknowledgments (ACKs), according to the ACK policy specified in the MAC header of each frame. The standard encompasses three different ACK policies, namely *No ACK*, *Immediate ACK*, *Delayed ACK*.

Selecting the *No ACK* option in the frame header, the destination is not required (nor allowed) to return any acknowledgement for that frame. The *Immediate ACK* option, instead, instructs the destination to return an ACK packet within a given time interval (Short Inter-Frame Spacing) after the successful reception of the frame. Finally, the *Delayed ACK* option allows the source to send a given number of frames before waiting for a cumulative ACK that fix which frames have been correctly received. Such an option, specifically designed for isochronous connections, has to be negotiated by the peer DEVs before the communication can take place.

Management–CTAs (MCTAs) share the same structure of CTAs, with the difference that they are reserved for communications involving the PNC, so that DEV–to–DEV communication are not allowed in MCTAs. The standard encompasses three types of MCTAs, namely *Regular*, *Association*, and *Public*. *Regular* MCTAs are univocally assigned to specific DEVs and can be used for different purposes, as exchanging control information with the PNC in a collision–free manner. *Association* and *Public* MCTAs, instead, are not reserved and can be used by any DEV through a slotted ALOHA channel access procedure. *Association* MCTAs are dedicated to association and disassociation requests by the DEVs, whereas *Public* MCTAs can be used for exchanging generic control messages, as CTA allocation requests. As usual, the beacon frame at the beginning of the superframe specifies the position, duration and type of each MCTA, as well as the DEV associated to each regular MCTA.

The subdivision of the frame in CAP and CTAP, and the assignment of CTAs and MCTAs to the different DEVs, is univocally determined by the PNC on the basis of the amount of resources requested by each DEV in the piconet. Although the standard defines the basic mechanisms that permit the exchange of information between PNC and DEV, it does not specify how the info exchange shall be realized, nor the way such information has to be used.

III. PRIOR WORK

Prior work on this area is rather scarce. To begin with, [2] proposes a rate adaptation mechanism for HR WPAN, whereas a scheme to recover the bandwidth waste due to imperfect scheduling is presented in [3].

The issue of managing different priority traffic classes in saturated networks is considered in [4]. More specifically, the authors focus on a scenario where transmission resources are statically assigned to traffic flows, upon request. According to the standard resource–management policy encompassed by IEEE 802.15.3 specs [5], a request that exceeds the amount of available resources is rejected, irrespective of its priority class. Therefore, a high priority flow request might be delayed because transmission resources are fully occupied by lower priority flows. To alleviate such a problem, [4] proposes a novel scheduling scheme that reduces the access delay of high priority flows by "stealing" some transmission resources from lower priority flows. In this manner, the system is able to provide fast channel access to high priority flows at the cost of slowing down the service offered to lower priority flows.

The throughput maximization and delay minimization in HR WPAN is addressed in [6], where a scheme to manage multimedia traffic at MAC layer is presented. The basic idea is to balance the amount of resources dedicated to contention–based and contention–free access, according to the overall offered traffic. In fact, contention–based access is more efficient under light traffic loads, whereas contention–free access becomes preferable when the traffic load increases. These works refer to scenarios with traffic belonging to different priority classes, but they do not specify nor consider the actual nature of the traffic.

A step forward in this direction is made in [7], where the authors explicitly address the problem of transmitting MPEG4 coded video flows over a high rate WPAN and propose a resource allocation that keeps into consideration the typical traffic–pattern of MPEG4 sources. However, to reach optimal performance, the scheme needs to know in advance the characteristics of the MPEG4 flow (as the structure of the frame–stream and the maximum size of each frame type): imperfect knowledge yields to performance loss. Moreover, a portion of the allocated bandwidth remains unused any time the transmitted frame is smaller than its maximum expected size.

This problem is alleviated by the FACTA scheduling scheme, proposed in [8], which dynamically adapts the resource allocation on the basis of the feedback provided by the DEVs to the PNC. The scheme exploits the characteristics of MPEG stream, performing channel–time allocation in accordance with the pre–assigned priority of each MPEG frame type.

Another solution for providing QoS in WPAN is presented in [9x], where authors resort to traditional control feedback theory for controlling the backlog of the different flows. Transmission resource is, then, allocated in order to stabilize the queue length of each flow around a desired target value that is related to the required QoS.

All the above schemes, however, have been proposed and analyzed in scenarios with homogeneous traffic sources and they may lead to inefficient resource usage in presence of traffic flows with different QoS requirements. To the authors' knowledge, the current literature still lacks the performance analysis of HR-WPAN in more realistic scenarios where traffic sources of different nature have to coexist. The purpose of this work, therefore, is to shed light on this aspect.

IV. Scheduling Algorithms

In this section we describe the scheduling algorithms considered in the study. Instead of designing new schemes from scratch, we prefer considering off the shelf solutions that might be readily deployed in commercial devices with minimal implementation effort. Along this line of rationale, we assume a very simple QoS model, which takes into account only delay constraints. we distinguish between Hard–QoS applications, which suffer strong quality degradation in case of delay–bound violation, and Soft–QoS applications, which show progressive service degradation as the delay increases over the soft deadline. In addition, Soft–QoS applications can be associated to a second hard deadline, after which packet delivery is useless.

The scheduling algorithms here considered, therefore, will make use (at most) of the following stream parameters:

- **Channel Time Request** (*CTRq*). Time required by the DEV to complete a *task*, which consists in the successful transmission of the pending data for that stream.
- **Hard Timeout**. Time left before the task hard deadline is reached.
- **Soft Timeout**. Time left before the task soft deadline is reached.

These parameters are collected by the PNC once per superframe, through the service registration and polling schemes that will be described in Section V. Then, they are fed to the scheduling algorithm that determines the resource assignments for the upcoming superframe. The schedule is, hence, broadcasted in the beacon message at the beginning of the superframe.

Out of the plethora of classical solutions available in literature, we selected four well–known algorithms, namely *Generalized Processor Sharing* (GPS) and *Early Deadline First* (EDF) that comes along with two evolutions: *EDF with Discard* (EDF–DS) and *EDF with Soft/Hard deadlines*. The algorithms are described in greater detail in the following.

A. Generalized Processor Sharing (GPS)

Generalized Processor Sharing (GPS), together with its numerous variants [10x], [11x], is widely used in packet network scheduling and CPU processes management. It has been selected as representative of the large family of fair–share scheduling algorithms, which also includes Weighted Round Robin, Deficit Round Robin, and others. These algorithms work on the resource demand of the different streams, without considering the QoS constraints. We decided to include one of such algorithms in our analysis in order to have a term of comparison with QoS–oriented algorithms.

A GPS scheduler only needs to know the *CTRq* for each DEV. More specifically, denoting by T the allocable time in a superframe and by $CTRq_i$ the time requested by the i–th stream, $i = 1, 2, \ldots, n$, then the time interval ideally assigned to the i–th stream is given by

$$\tau_i = T\frac{CTR_{q_i}}{\sum_k CTR_{q_k}} \tag{1}$$

In practice, however, DEVs are not capable of using any arbitrary time allocation, since the transmission time of an elementary data unit is constrained by several factors, such as fragmentation and reassembly scheme, frame size, physical layer transmission rate, acknowledgment policy. Therefore, a practical realization of GPS has to take into account these significant restrictions in order to limit as much as possible the potential resource waste due to unfeasible resource allocations.

In this work, we assume that, at the connection time, each DEV communicate to the PNC its Time Unit, i.e., the minimal unit of channel time the DEV is able to use in a profitable manner. The GPS scheduler, hence, approximates the theoretical time-share τ_i with the closest multiple of the DEV's Time Unit. In any case, the computation of the resource–share is trivial and can be performed with very limited memory and computational resources.

B. Earliest Deadline First (EDF)

As representative for the delay–aware scheduling algorithms, we considered the Earliest Deadline First (EDF) algorithm, which is one of the most known real–time algorithms with dynamic priority [12]. In a single–hop context (as in a 802.15.3 piconet) EDF represents an optimal scheduling policy: if a set of tasks is completely allocable under whichever discipline, then it is certainly allocable also under EDF [13]. More specifically EDF maximizes the service admission region for traffic classes with different deadlines.

The EDF scheduler needs to know the current *CTRq* and *Hard Timeout* parameters for each stream. The resource scheduling is, hence, performed in two steps. First, the EDF scheduler sorts the stream requests in ascending order of *Hard Timeout* threshold. Second, the scheduler assigns the requested *CTRq* to each stream, starting from the first of the list (which has the closest timeout), till requests are all satisfied or resources run out. Therefore, EDF scheduling requires more information than GP and involves a list–sort operation, whose computational load is limited given the rather small number of simultaneous streams that can be realistically expected in the system.

The EDF approach is very effective from a real–time perspective and it is the progenitor of more advanced algorithms. In this work, beside basic EDF, we consider two evolutions, i.e., EDF with Discard (EDF–DS) and EDF with Soft/Hard deadlines (EDF–SH), which are described in the following.

C. Earliest Deadline First with Discard (EDF–DS)

EDF–DS enriches the original EDF scheduler by including this *discard* policy. In fact, the basic version of EDF makes use of the hard timeout threshold only to determine the task priority in the service list: tasks that are closer to their timeout get served first. However, no check is made to verify whether the task service will be concluded within the timeout deadline. Hence, a task can be served after that its hard deadline has passed, thus resulting in a waste of resources. This drawback can be avoided by a preventive control on the task deadline: if a given task cannot be served within its timeout then the task is given no resources (*discard*).

In literature several discard policies, as well as medium access rearrangements, have been proposed and analyzed [12], [14]. In the context of HR–WPANs, however, simplicity is often preferable to a marginal performance gain that might be achieved by adopting more complex solutions. Hence, in this study we consider a simple EDF–DS strategy, in which a packet is discarded when its hard timeout cannot be met at the EDF–assign time. In addition to the operations performed by the original EDR scheme, EDR–DS requires performing only two basic operations (sum and comparison) for each resource request, so that the complexity of the two algorithms is basically the same.

D. Earliest Deadline First, Soft/Hard Deadlines (EDF–SH)

A further evolution of the basic EDF approach is the EDF–SH, which attempts to offer a privileged service to traffic characterized by both soft and hard deadlines, without severely affecting the other content. EDF–SH, in fact, attempts to serve the tasks before they meet their Soft Deadlines (admission in advance), provided that no other streams suffer job failures because of this.

The scheduler sorts the allocation requests in ascending order of *Soft Timeout*. Then, channel time requests (*CTRq*) are dispatched, starting from the first request of the list. If a violation occurs in the allocation of the i–th request, i.e., task i would not complete its service before its *Hard Timeout*, then the scheduler makes a backward search to verify which of the already allocated tasks can be delayed of $CTRq_i$ without violating their hard timeout constraint. If k of these streams can be deferred, then the i–th stream is anticipated of k positions in the service list, in order to meet its deadline. Of course, in case the research fails, the usual discard policy takes place.

Computationally, the EDF–SH algorithm is more demanding than EDF and EDF–DS, due to the need of backward list searching and reordering. In the worst case, the EDF–DS scheme may require order of n^2 basic operations, where n is the total number of simultaneous streams, in addition to those performed by original EDR scheme. However, being n limited to few tens, the complexity is still manageable by modern processing units.

V. Polling Procedure

As previously mentioned, the scheduling algorithms need a mechanism to collect QoS requirements from the DEVs. To this end, we introduce a customized polling procedure that can be realized in a completely standard–compliant way by leveraging on the set of tools provided by the 802.15.3 specifications. The polling procedure makes use of two customized control messages, referred to as *commands*, which we have named *Stream Properties Command* and *Sender Status Command*. (Customized commands can be realized by using the class of Vendor Specific Command encompassed by the standard.)

The structure and use of these commands are described below.

A. Stream Properties Command: connection setup

Before transmitting data, a DEV needs first to register its new stream with the PNC, as encompassed by the IEEE 802.15.3 specifications. To this end, the DEV sends a suitable command to the PNC by using the *Association MCTAs* allocated by the PNC in the current superframe (see Section II). The PNC replies to the DEV command by assigning a *Stream ID* sequence (8 bits) to the data stream and allocating a regular MCTA in each successive superframe for DEV–PNC communications. In order for the scheduling algorithm to operate upon as much fresh info as possible, the MCTAs are allocated at the end of each superframe, as depicted in Fig. 1.

To gain access to the advanced scheduling scheme provided by the PNC, the DEV needs to acquaint the PNC with the QoS requirements of its stream. To this purpose, it sends a *Stream Properties Command* in the assigned MCTA, which carries the following fields:

- *Stream ID*. Stream identifier (8 bits).
- *Hard Deadline Value*. Hard deadline constraint for the data stream (8 bits), resolution 1 ms, range [0÷255] ms.

- *Soft Deadline Value*. Soft deadline constraint for the data stream (8 bits), resolution 1 ms, range [0÷255] ms.

If the Soft Deadline constraint does not apply for that particular type of stream, the field is set equal to the Hard Deadline Value.

The Stream Properties Command is sent only once, when the stream is first instantiated. Therefore, its impact on the system performance is negligible.

B. Sender Status Command: periodic polling.

Once the stream is established, the PNC needs to monitor the status of the ongoing flow, in order to update accordingly the parameters used by the scheduling algorithm. To this end, each DEV periodically sends to the PNC a *Stream Status Command*, by using the assigned regular MCTA. Such a command is obtained by adding a 64–bit MAC header to the concatenation of a number of basic structures, called *Stream Status Blocks*, each associated to an active stream originating from the DEV. A Stream Status Block consists of 32 bits organized as follows:

- *Stream ID*. Stream identifier (8 bits).
- *Channel Time Request* (*CTRq*). Time required by the DEV to transmit the pending data (16 bits), resolution 1μs, range [0÷65535] μs. (The transmission of such data represents a task.)
- *Waited Time*. Amount of time already consumed in queue by the current stream task (8 bits), resolution 1ms, range [0 ÷ 255] ms).

The *Stream ID* field is that assigned to the data stream when it was instantiated, as reported in the Stream Properties Command. The *CTRq* field is determined according to the amount of data the current task is composed of, the physical rate used by the DEV, and the ACK policy implemented by the DEV. The *Waited Time* field, finally, is updated by the DEV any time a new Stream Status Command is sent.

Subtracting the *Waited Time* fields from *Hard* (*Soft*) *Deadline Value* declared in the Stream Properties Command, the PNC determines the current value of the hard (soft) timeout for the considered task. The scheduling algorithm in the PNC uses the information delivered through these commands to adjust the resource allocation in the successive superframe.

It might be worth remarking that the described procedure could be implemented in several different ways and the solution here proposed does not claim in any way to be the best possible. Nonetheless, it takes the benefit of simplicity and adherence to the standard.

VI. Simulation Settings

The performance analysis presented in the following has been carried out by using a WPAN simulator written in C++. The simulator includes the basic functionalities of the standard and a detailed implementation of the Frame Convergence Sublayer (FCSL), Medium Access Control (MAC) and Physical (PHY) layers, following the specifications contained in [5] for the 2.4 GHz band. The scenarios and parameter settings used in the simulations are described below.

A. Multimedia Traffic Models

In the effort to compare the scheduling algorithms in a challenging and realistic scenario, we considered both Hard– and Soft–QoS sources. More specifically, we focused on MPEG4, Voice over IP and interactive–gaming traffic flows.

1) Audio/Video Traffic (MPEG-4)

An MPEG4 flow is a continuous stream of audio/video frames, emitted at a constant frame rate. Frames can be intra–coded (I), predicted (P) from a previous I–frame, or bi-directionally predicted (B) from previous and next I/P frames. Accordingly, I–, P– and B–frames have progressively smaller size. The video stream is structured in predefined frame sequences, called Group of Pictures (GoP), e.g., I-B-B-P-B-B-P-B-B-P-B-B. Therefore, the bit rate generated by the encoder is highly variable, depending on both the characteristics of the audio/video content and the GoP structure. For a more detailed description of the codec, which is out of the scope of this paper, we refer to [15], from which we have also taken the following traffic:

- Hard–QoS requirements (Hard Deadline: 40ms);
- Constant frame rate: 25 frames per seconds;
- GoP: I-B-B-P-B-B-P-B-B-P-B-B
- Average bit rate: 1.33 Mb/s;
- Simulated transport protocol: UDP.

2) Voice Over IP Traffic (VoIP)

Several different voice codecs have been proposed in the last decade. In this work, we consider G.723.1 high bit rate encoders Multi-Pulse - Maximum Likelihood Quantizer (MP-MLQ), which generate packets of 24 bytes every 30 ms. Each packet goes through RTP, UDP and IP protocol layers before reaching the 802.15.3 FCSL layer, so that the packet to be transmitted is 64 bytes long.

To simulate talk and silence periods of voice conversations, we adopt the classical two–state (ON–OFF) Markov model, with average ON and OFF periods of 352 ms and 650 ms, respectively [16]. In ON state, the source generates voice packets at a constant rate, according to the voice model just described, whereas in OFF state the source is idle and no data is generated (silence suppression). According to the International Telecommunication Union Recommendation G.114, a delay lower than 150 ms is acceptable for most user applications. Hence, we classify the VoIP as hard QoS application, with deadline fixed to 100 ms, in order to leave some margin for extra delay introduced by the path external to the WPAN.

The VoIP model is summarized as follows:

- Hard–QoS requirements (Hard Deadline: 100ms);
- Markovian ON/OFF model (voice activity/silence);
- Constant frame rate: 33 frames per seconds (during ON periods);
- Average bit rate: < 20 kbps;
- Simulated transport protocols: RTP over UDP.

3) Interactive Gaming Traffic (i–gaming)

Another class of applications that is expected to become very popular in the WPAN community is Interactive Gaming (*i*–gaming). A typical *i*–gaming scenario encompasses a rather large number of clients connected to a single game server, which receives and re–transmits all the information necessary for organizing the collaborative game. I–gaming is a good example of soft QoS applications: when packet delay exceeds a soft deadline, the user experience smoothly degrades with the increasing of the packet delivery delay, up to a threshold (corresponding to the Hard Deadline) beyond which the game session will likely be quitted by the player.

TABLE I
I-GAMING, MODEL

Channel	Packet Size	Inter-arrival time
Uplink	Gamma ($\alpha = 29.8$, $\beta = 1.5$)	Rayleigh ($\beta = 1.5$)
Downlink	Gamma ($\alpha = 4.2$, $\beta = 29$)	Rayleigh ($\beta = 1.5$)

To model this source, we have proceeded to direct inspection of long sequences of traffic generated by a First Person Shooter, with tight delay and rate bounds.

We have collected packet size and inter–arrival time for a number of different sessions of 20 minutes each, all with 20 players from different domains and we have observed that the empirical statistical distribution of the packet size and inter–arrival time could be fairly well matched by Gamma and Rayleigh distributions, respectively, with parameters as in Table I. This model is in good agreement with some other models proposed in literature [17].

The Soft and Hard deadlines have been set to 15 ms and 30ms, respectively, which represent a rather critical, though realistic, scenario considering the possible presence of network paths external to the piconet. In summary, *i*–gaming traffic is modeled as follows:

- Soft QoS requirements (Soft Deadline: 15 ms; Hard Deadline: 30 ms)
- Variable Packet Rate
- Independent uplink and downlink traffic
- Simulated transport protocol: UDP.

B. Performance Measure

As in [7–9], we borrow some definitions from the computer science terminology. We say that a generic stream undergoes a job failure whenever a data packet is not completely or correctly delivered within the associated hard deadline: the job failure rate (JFR) will measure the frequency of such an event. System performance is, hence, evaluated in terms of JFR. We say that the piconet offers high QoS when each stream experiments a JFR less than 1%, on the average.

C. Beacon, CAP and Superframe Duration

According to the literature, the collision–based access mechanism used in CAP is not suitable for supporting intense multimedia traffic [4], [6], [7]. Hence, we focus on the CTAP only. Nonetheless, we maintain a short CAP to allow the association of new DEVs that wish to join the piconet. The duration of CAP is scaled with the size of the beacon frame, in such a way that the overall time occupancy of beacon and CAP remains fixed to 300 μs, irrespective of the number of active streams. In this way, we provide a sort of control over the total number of connections that can be simultaneously active in the piconet. In fact, the greater the number of active streams and, in turn, the size of the beacon frames, the lower the chance for a new DEV to join the piconet.

Since CTAs allocation is announced in the beacon at the beginning of the superframe and not changed until the successive superframe, the superframe duration has a crucial impact on the piconet performance. Short superframes permit fine–grain resource allocation, easier error recovery, and reduced channel access delay. On the other hand, long superframes yield overhead reduction and, therefore, better resource usage. In this study, we fixed the superframe duration to 10 ms, which proved to be a rather good compromise according to the results of a preliminary study, here omitted for space constraints. This superframe setting is reasonable for a rather wide range of different traffic classes, though specific objectives might suggest different values.

D. Simulation scenarios

Simulations have been carried out in two different application scenarios referred to as *homogeneous* and *heterogeneous*. In the homogeneous traffic scenarios, we have considered only MPEG4 flows, in order to evaluate the potentialities of 802.15.3 WPAN in supporting this specific (and critical) type of traffic. In the heterogeneous scenarios, instead, we have considered a mix of different applications, in order to compare the scheduling algorithms in presence of heterogeneous QoS requirements and traffic patterns.

The transmission rate of each DEV has been randomly selected in the set of admissible physical rates, i.e., {22, 33, 44, 55} Mb/s, with probability {0.1, 0.2, 0.3, 0.4}, respectively. The average transmission rate is 44 Mb/s. Unless differently stated, the Frame Error Rate (FER) has been set to 4% for a reference frame payload length of 2044 bytes. Frames are immediately acknowledged by the receiver (Immediate ACK policy.)

Each simulation lasted for 20 (virtual) minutes, starting from the time all the DEVs were connected to the PNC.

Notice that, for clarity, we do not plot the EDF–SH curves when the simulated scenario does not include any soft–QoS applications, since in this case it behaves as the EDF–DS algorithm.

VII. SIMULATION RESULTS

In this section we report the simulation results obtained in the different scenarios. We first describe the results obtained in the case of homogeneous traffic. Then, we present and discuss the performance achieved by the scheduling algorithms in the more challenging case of mixed traffic.

A. Homogeneous Traffic

The simulations of homogeneous traffic scenarios have been performed by progressively increasing the number of active MPEG4 flows.

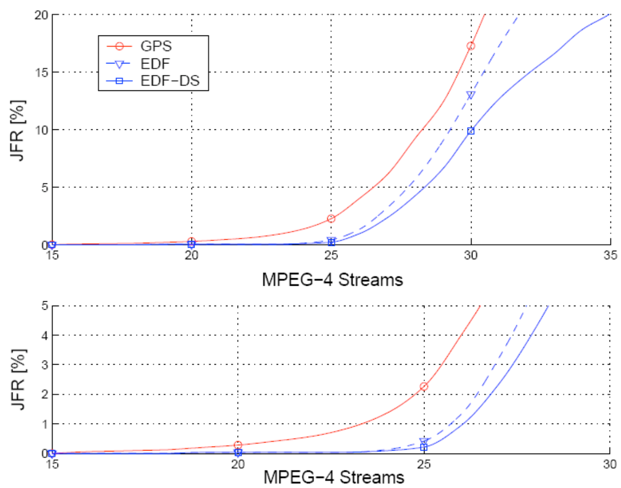

Fig. 2. MPEG-4: JFR vs. number of MPEG-s streams (above) and zoom in the region with JFR≤5% (below). FER=4%

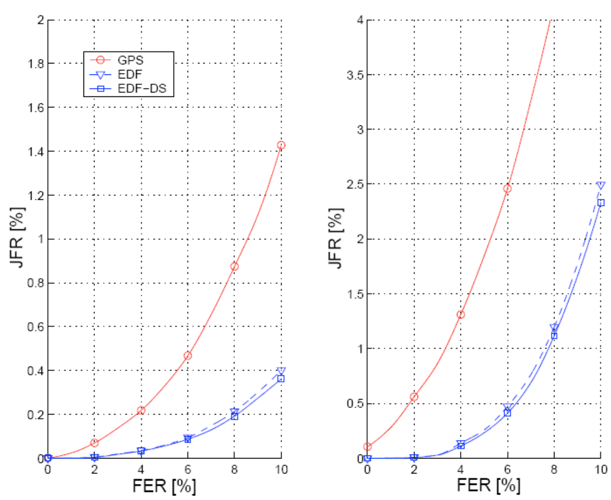

Fig. 3. MPEG-4: JFR vs. FER (%)under moderate (left-hand side) and high (right-hand side) traffic load.

In Fig. 2, we show the JFR curve realized by GPS, EDF and EDF–DS scheduling algorithms. (The lower graph zooms in the region with JFR ≤ 5%).

At first, we notice that all the scheduling algorithms achieve similar and, in general, rather good performance, being able to support up to 20 MPEG4 streams with less than 2% of JFR.

However, EDF and EDF–DS algorithms perform slightly better than GPS, enlarging the high–QoS region as shown on the bottom graph of Fig. 2. The graphs also show that JFR remains almost constant when the traffic increases up to a given threshold, after which the JFR sharply increases. This behavior confirms that the resource management is very effective, being able to satisfy the traffic requirements until the offered load gets close to the saturation threshold. In Fig. 2 it is also possible to appreciate as the discard/recovery mechanism implemented by EDF–DS can bring some performance improvement over the simple EDF. Nonetheless, such an improvement is not significant in the region of interest for practical usage. On the contrary, GPS is less efficient in dealing with temporary traffic overloads, thus showing a contained but progressive quality loss when traffic load increases.

Fig. 3 shows the impact of FER on JFR for the three scheduling algorithms. The left–hand graph plots the results obtained with moderate traffic, i.e. with 20 MPEG4 streams that correspond to approximately 60% of channel occupancy.

The right–hand graph shows the curves for high traffic, with 30 MPEG4 streams and approximately 90% of channel occupancy. In both cases, the overhead due to the polling scheme and corresponding to the allocation of the MCTA slots at the end of the superframe, was negligible (<10 μs).

The figure clearly shows that the priority–based admission criterion used by EDFs algorithms has a positive effect on the system performance. In fact, frames undergoing retransmissions are likely to be rescheduled by EDF algorithms before frames with more relaxed time constraints. On the contrary, GPS entails noticeable performance worsening when the traffic load increases, since the scheduling policy does not take into account any time constraints.

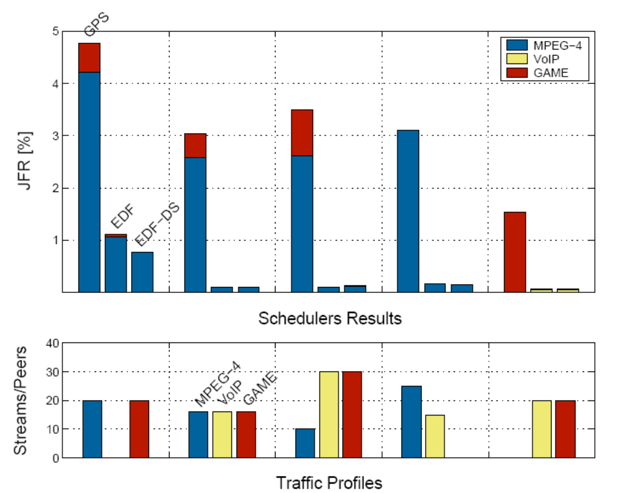

Fig. 4. Cumulative JFR for GPS, EDF, and EDF-DS (above) in scenarios with traffic mix as reported below. FER=4%.

B. Heterogeneous Traffic

In a heterogeneous traffic scenario, the scheduling algorithm faces a new challenge: besides maximizing the number of sustainable flows with satisfactory QoS, the scheduler is also asked to provide fairness among different traffic classes. To investigate these two aspects, we have performed two distinct simulations series.

In the first simulation campaign, we have considered five traffic profiles, each obtained by mixing MPEG4, VoIP and *i*–gaming sources in the percentages reported below in Fig. 4. To challenge the scheduling algorithm, we have imposed a high traffic load. Above in Fig. 4 we report the JFR obtained for each traffic class with the different scheduling algorithms. (EDF–SH results have been omitted since they basically overlap with EDF–DS ones.) Each group of bars is referred to the traffic mix reported in the immediately underneath graph. Notice that, with the right–most traffic profile (20 VoIP full–duplex streams plus 20 *i*–gaming peers), we limited to 40% (4 ms) the amount of allocable resources, in order to produce some appreciable results in a reasonable simulation time. At a first glance, we observe that the JFR for MPEG4 obtained by adopting the GPS scheduler, which was comparable with that obtained by EDF schedulers in the homogeneous traffic scenarios with only MPEG4 sources, now shows a remarkable decay, being 8÷16 times higher than that obtained with EDF algorithms. This is expected since GPS does not take into consideration the residual lifetime of data packets, but only the *CTRq* values declared by the sources. Therefore, GPS privileges sources with low bandwidth demand, such as VoIP, to those having more stringent delay requirements.

In heterogeneous scenarios, in fact, the awareness of tasks deadline plays an essential role. Hence, scheduling algorithms that make use of this information, such as EDFs, are able to maintain the service in a high QoS region even where other policies yield practically unacceptable performance.

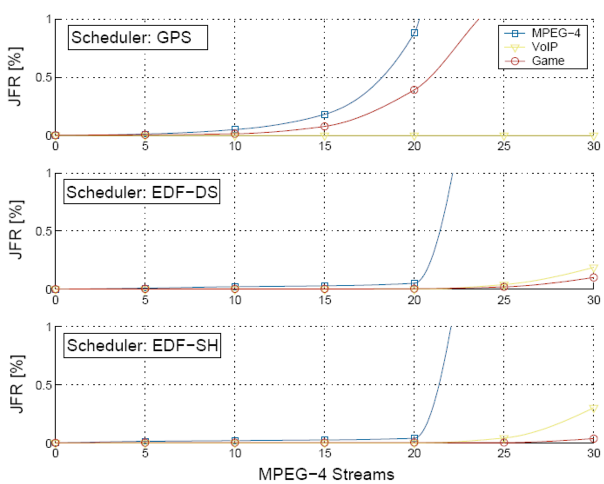

Fig. 5. JFR per traffic class measured in a moderately loaded piconet, while increasing the number of MPEG4 transmissions.

In the second set of simulations, we have compared the different scheduling algorithms for increasing traffic load. More specifically, the piconet was initially interested by 16 voice and 16 *i*–gaming sessions, corresponding to a modest traffic load (approx 20% of channel occupancy). Hence, MPEG4 streams were progressively introduced. Fig. 5 depicts the JFR vs. the number of active MPEG4 flows for GPS, EDF–DS and EDF–SH. (For clarity, we omit pure–EDF results, which are in good agreement with what already observed in the previous cases.)

The results confirm the previous consideration on GPS and EDF–DS performance, as well as the substantial equivalence between EDF–DS and EDF–SH algorithms, though EDF–SH undergoes a very marginal penalization in the joined MPEG4/VoIP traffic, balanced by an extremely high QoS for *i*–gaming. Notice, that this fact is observed in a traffic region where MPEG4 streams are experiencing low quality levels, so that the scenario is unlikely to occur in practical cases. The EDF approach generally exhibits a threshold–behavior: the offered service is excellent until the traffic load approaches a critical level, after which performance undergoes rapid deterioration.

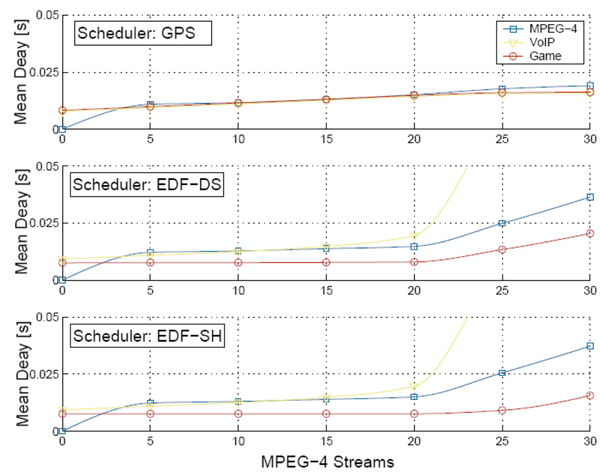

Fig. 6. Delay per traffic class measured in a moderately loaded piconet, while increasing the number of MPEG4 transmissions.

Fig. 6 shows the mean delay values measured in the same test conditions. Notice that the delay is obtained only for the packets that are successfully delivered to the destination, so that the results shown in Fig. 6 are significant only in the region where the JFR is acceptable. We observe that GPS inevitably undergoes a generalized progression in the performance decay, whereas EDFs schemes are able to offer a sort of isolation between different classes.

It is also interesting to notice that, in low or moderate load conditions, EDF–SH does not provide any appreciable latency gain on EDF–DS, in spite of its more complex scheduling mechanism. Results become more favorable to EDF–SH for high traffic loads, a condition that, however, is unlikely to hold in practical cases.

VIII. Conclusions

In this work we have compared some possible scheduling algorithms for providing high QoS to multimedia traffic in 802.15.3 piconets. To this end, we have proposed a simple and standard–compliant polling mechanism for the piconet coordinator (PNC) to gather traffic information from the DEVs. Hence, we have performed several simulations in different conditions, to explore pros and cons of the considered scheduling algorithms.

The analysis has privileged applicative scenarios, in which all the algorithms have proved to be able to provide high QoS service to different type of applications.

In general, EDF–based algorithms have shown better performance than GPS, with larger gain margin in heterogeneous traffic scenarios. Furthermore, EDF–DS and EDF–SH have shown some performance gain on pure EDF, whereas the higher complexity of the EDF–SH mechanism has not proved to pay enough in terms of performance improvement with respect to EDF–DS, so that the last algorithm seems to be preferable in almost all the cases.

In conclusion, EDF–based schedulers are able to provide high QoS in presence of heterogeneous multimedia traffic flows. To reach this goal, however, EDF algorithms need to get access to some cross–layer information, such as the hard and soft timeout, which depend on the application. Therefore, performance gain might be expected from an optimization of the polling and signaling procedures that realize the cross layer interaction.

REFERENCES

[1] H. Shinde and M. Borse, “High-rate wireless personal area networks,” in *IEEE International Conference on Personal Wireless Communications, 2005*. ICPWC 2005., 23-25 Jan., 2005, pp. 19–23.

[2] B. Kim, F. Yuguang, and T. Wong, “Rate-adaptive MAC protocol in high-rate personal area networks,” in *IEEE Wireless Communications and Networking Conference (WCNC 2004),* vol. 3, 21-25 March 2004, pp. 1394– 1399.

[3] E. Kwon, D. Hwang, and J. Lim, “An idle timeslot reuse scheme for IEEE 802.15.3 high-rate wireless personal area networks,” in *IEEE Vehicular Technology Conference, 2005 (VTC-2005-Fall)*, vol. 2, 25-28 Sept., 2005, pp. 715– 719.

[4] R. Zeng and G.-S. Kuo, “A novel scheduling scheme and MAC enhancements for IEEE 802.15.3 high-rate WPAN,” in *IEEE Wireless Communications and Networking Conference (WCNC),* vol. 4, 13-17 March 2005 2005, pp. 2478 – 2483.

[5] IEEE, “IEEE std 802.15.3, part 15.3: Wireless medium access control (mac) and physical layer (phy) specifications for high rate wireless personal area networks (wpans),” 2003.

[6] R. Fantacci, D. Tarchi, and G. Izzo, “Multimedia traffic management at MAC layer in IEEE 802.15.3a personal area networks,” in *International Conference on Wireless Networks, Communications and Mobile Computing*, Kemer, Turkey, July 2005, pp. 923–928.

[7] S. H. Rhee, K. Chung, Y. Kim, W. Yoon, and K. S. Chang, “An application-aware MAC scheme for IEEE 802.15.3 high-rate WPAN,” in *IEEE Wireless Communications and Networking Conference (WCNC),* 2004, pp. 1018 – 1023.

[8] Sun-Myeng Kim and Young-Jong Cho “Scheduling Scheme for Providing QoS to Real-Time Multimedia Traffics in High-Rate Wireless PANs,” *IEEE Transactions on Consumer Electronics*, Vol. 51, No. 4, NOVEMBER 2005

[9] X. Liu, Q. Dai, and Q. Wu, “Link-level scheduling for providing QoS in WPAN,” *in International Conference on Communications, Circuits and Systems Proceedings*, vol. 2, June 2006, pp. 1372–1376.

[10] N. Bansal and M. Harchol-Balter, “Analysis of SRPT scheduling: investigating unfairness,” in *SIGMETRICS/Performance, 2001*, pp. 279–290.

[11] A. Parekh and R. Gallager, “A generalized processor sharing approach to flow control in integrated services networks: The single-node case,” *IEEE/ACM Trans. Networking*, Jun 1993, pp. 344–357.

[12] V. Sivaraman, F. M. Chiussi, and M. Gerla, “End-to-end statistical delay service under GPS and EDF scheduling: A comparison study,” in *Proceedings of INFOCOM*, 2001, pp. 1113–1122.

[13] Liebeherr, D. E. Wrege, and D. Ferrari, “Exact admission control for networks with a bounded delay service,” *IEEE/ACM Transactions on Networking*, vol. 4, no. 6, pp. 885–901, 1996

[14] Y. Xie and T. Yang, “Cell discarding policies supporting multiple delay and loss requirements in atm networks,” *in Proceedings of GLOBECOM*, Phoenix, AZ, September 1997, pp. 1075–1080.

[15] F. H. Fitzek and M. Reisslein, “Mpeg-4 and h.263 video traces for network performance evaluation,” Technical University Berlin, Berlin, TKN, Technical Report, October 2000.

[16] P. Fiorini, “Voice over ip (voip) for enterprise networks: Performance implications and traffic models,” *BMC Software*, Tech. Rep., 2001J..

[17] Johannes Färber, “Network Game Trffic Modelling,” in *NetGames '02: Proceedings of the 1st workshop on Network and system support for games.* New York, NY, USA: ACM Press, 2002, pp. 53-57.

Andrea Zanella is Assistant Professor at the Department of Information Engineering (DEI), University of Padova (ITALY). He received the Laurea degree in Computer Engineering in 1998, from the same University, and the PhD degree in Electronic and Engineering, in 2002. Before that, he spent nine months as visiting scholar at the Department of Computer Science of the University of California, Los Angeles (UCLA), where he worked with Prof. Mario Gerla on Wireless Networks and Wireless Access to Internet. Andrea Zanella's major research interest is in the field of protocol design and performance evaluation of WPANs, MANETs, VANETs and WSNs. He his IEEE member and reviewer for several IEEE journals and international conferences in the ICT area.

Fabio Lorquando has received the Laurea degree in Telecommunications Engineering in 2006, thereafter actively collaborating with the Department of Information Engineering (DEI), University of Padova (ITALY). Fabio Lorquando's major research interests are in the fields of WPANs performance evaluation and compiling technologies.