Combining Distributed Queuing with energy harvesting to enable perpetual distributed data collection applications

Francisco Vazquez-Gallego[1,*], Pere Tuset-Peiró[2], Luis Alonso[3], Jesus Alonso-Zarate[1]

[1]Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), Castelldefels, Spain.
[2]Universitat Oberta de Catalunya (UOC), Barcelona, Spain.
[3]Universitat Politècnica de Catalunya (UPC), Castelldefels, Spain.

ABSTRACT

This paper presents, models and evaluates EH-DQ (Energy Harvesting-aware Distributed Queuing), a novel MAC protocol that combines Distributed Queuing (DQ) with Energy Harvesting (EH) to address data collection applications in industrial scenarios using long-range and low-power wireless communication technologies. We model the MAC protocol operation using a Markov chain and evaluate its ability to successfully transmit data without depleting the energy stored at the end-devices. In particular, we compare the performance and energy consumption of EH-DQ with that of TDMA (Time Division Multiple Access), which provides an upper limit in terms of data delivery, and EH-RDFSA (EH-aware Reservation Dynamic Frame Slotted-ALOHA), which is an improved variation of FSA (Frame Slotted ALOHA). To evaluate the performance of these protocols we use two performance metrics: the delivery ratio and the time efficiency. The delivery ratio measures the ability to successfully transmit data without depleting the energy reserves, whereas the time efficiency measures the amount of data that can be transmitted in a certain amount of time. Results show that EH-DQ and TDMA perform close to the optimum in terms of data delivery, and both outperform EH-RDFSA in terms of data delivery and time efficiency. Compared to TDMA, the time efficiency of EH-DQ is insensitive to the amount of harvested energy, making it more suitable for energy-constrained applications. Moreover, compared to TDMA, EH-DQ does not require updated network information to maintain a collision-free schedule, making it suitable for very dynamic networks.

***Correspondence**

Computer Science, Multimedia and Telecommunication Department, Universitat Oberta de Catalunya. Rambla del Poblenou 156, 08018 Barcelona, Spain. E-Mail: peretuset@uoc.edu.

1. INTRODUCTION

Distributed data collection is an important use case in the upcoming fourth Industrial revolution, as new information will need to be extracted from equipment or machines deployed within cities and factories in order to automate or optimize processes. In a distributed data collection application a gateway periodically requests data from end devices and, upon request, all end devices reply with their presence and the latest information available. Such networks are dynamic in nature and are formed by hundreds or thousands of devices, which are typically powered using batteries and communicate with the gateway using long-range and low-power wireless communication technologies.

A root cause of energy expenditure in such scenarios can be traced back to the MAC (Medium Access Control) layer [1], which coordinates the access to the wireless channel and determines when the radio transceivers are powered on to transmit, listen, or receive. There are two complementary strategies to prolong the network lifetime: *i)* reduce the energy consumption devoted to communications by means of reducing packet collisions, packet overhearing, idle listening and protocol overhead [2], [3], and *ii)* use energy harvesting systems that collect the needed energy from the environment [4], [5].

Motivated by this need, in this paper we focus on data collection scenarios where hundreds or thousands of end-devices, equipped with energy-harvesters, periodically transmit a burst of data packets upon request from a gateway. In such scenarios, a simple Time Division Multiple Access (TDMA) scheme would allow every end-device to transmit without collisions. However, this would come at the cost of requiring an updated knowledge of the network topology in order to maintain a collision-free schedule. In terms of delay and energy consumption, this may be an expensive procedure in highly dense networks. Contrarily, random access protocols do not require topology knowledge and their simplicity makes them ideal for simple and low-cost end-devices, but the effects of collisions increase energy expenditure reduce the network reliability.

An alternative to deterministic and random access protocols to improve the performance of the network and the energy consumption of end nodes is Distributed Queuing (DQ). DQ ensures collision-free data transmissions and offers a near optimum performance independent of the traffic load and the number of end-devices in the network, completely avoiding congestion regardless of how intense is the data traffic offered to the network and the traffic pattern generated. The collision of data packets is avoided by separating access requests from the transmission of data. DQ dedicates a very short contention window for access requests where the contention is resolved using a tree-splitting algorithm [6]. Previous works related to DQ have analyzed throughput in steady-state assuming that the end-devices generate data packets following a random Poisson distribution. Under this type of traffic, DQ achieves maximum throughput when only 3 access request slots per frame are used, regardless of the number of contending end-devices.

However, to the best of our knowledge, the design and analysis of a MAC protocol based on DQ for wireless networks with energy-harvesting has never received attention. Despite the use of energy harvesting may ideally provide infinite lifetime from the energy perspective, it may not guarantee fully continuous operation due to the high variability and unpredictability of the energy harvesting process. Hence, an end-device may enter temporarily in energy shortage when the energy available is not enough for the operation of the end-device. This fact needs to be considered for the design of an energy-efficient MAC protocol based on DQ which also take the energy-harvesting process into account. This is the main motivation for the work presented in this paper, which is an extension of the work presented in [7] and [8], and aims to fill this gap with the following contributions:

1. we propose Energy Harvesting-aware Distributed Queuing (EH-DQ), a MAC protocol that combines the Distributed Queuing (DQ) access technology with Energy Harvesting (EH). EH-DQ is suitable for data collection networks where each end-device is equipped with an energy harvester and generates messages which have to be fragmented into small data packets to be transmitted along the wireless channel. In EH-DQ, the end-devices only become active to transmit data if the energy available is above a predefined threshold.
2. through the use of a Markov chain model and assuming an ideal wireless channel, we analyze the evolution of the energy available in the end-devices using EH-DQ and derive closed-form equations to evaluate its performance in terms of *data delivery ratio (DDR)* and *time efficiency*. The DDR measures the ability of the MAC protocol to successfully transmit data to the gateway without depleting the energy reserves of the end-devices, and the time efficiency measures the amount of data that can be transmitted in a certain amount of time.
3. we compare the performance of EH-DQ with that of the upper-bound of an ideal TDMA protocol, and with the performance of the EH-RDFSA protocol, which is based on the Dynamic Frame Slotted-ALOHA protocol typically used in data collection networks.

The remainder of this paper is organized as follows. In Section 2, we present the related work to MAC protocols and energy harvesting. In Section 3, we describe the system model. In Section 4, we summarize the operation of the EH-DQ, EH-RDFSA and TDMA protocols. In Section 5, we describe the energy consumption model of EH-DQ. In Section 6, we describe the performance metrics. In Section 7, we present the analysis of the performance metrics. Section 8 is devoted to evaluate the performance of EH-DQ and validate the accuracy of the analysis through comprehensive computer-based simulations. The performance of EH-DQ, EH-RDFSA and TDMA are also compared in Section 8. Finally, Section 9 concludes the paper.

2. RELATED WORK

In this section, we present the related work, which includes an overview of Distributed Queuing, Energy Harvesting-Aware MAC protocols and, finally, the Energy Harvesting process.

2.1. Distributed Queuing

DQ was first introduced as DQ Random Access protocol (DQRAP) [9] for cable TV distribution, and later adapted to different communication networks; to wired centralized networks (Extended DQRAP [10], Prioritized DQRAP [11]), satellite communications with long propagation delays (Interleaved DQRAP [12]), code-division multiple access for 3G cellular networks (DQRAP/CDMA [13]), Wireless Local Area Networks (DQCA [14]), cooperative communications (DQCOOP [15]), wireless ad hoc networks (DQMAN [16]), and body area networks (DQBAN [17]).

More recently, the work in [18, 19] proposes an energy model of DQ for data collection scenarios. The model is later validated experimentally in [20], which demonstrates that DQ can indeed provide a network performance that is independent of traffic load and the number of end-devices in the network. Finally, in [21] the authors demonstrate the use of DQ for active

RFID networks operating in the 433 MHz band. Both theoretical and experimental results show that DQ can also reduce energy consumption in more than 80% with respect to FSA.

2.2. Energy Harvesting-Aware MAC Protocols

The majority of work related to MAC protocols with energy harvesting focuses on slotted-ALOHA [22], Carrier Sense-Multiple Access (CSMA) [23] and Dynamic Frame Slotted-ALOHA (DFSA) [24]. In particular, the FSA and DFSA protocols have been proposed in the past [25, 26, 27] for data collection scenarios with energy harvesting where each end-device has just one data packet to transmit per request to the gateway. Results show that DFSA outperforms FSA in terms of delay when it is optimally configured, i.e., the frame length is adjusted to the number of contenders in every frame.

More recently, the Energy Harvesting-aware Reservation DFSA protocol (EH-RDFSA) was proposed by the authors in [28] to improve the performance of DFSA when the end-devices generate messages fragmented into small packets. Results show that EH-RDFSA outperforms DFSA by letting end-devices reserve the channel, thus avoiding the need to compete for the channel for each newly generated fragment of the same message. Unfortunately, in order to minimize the probability of collision, all these variants of slotted-ALOHA require to adapt dynamically the frame length by estimating the number of contending end-devices, which may be difficult in highly dense networks where the end-devices may fall eventually in energy shortage.

2.3. Energy Harvesting process

Regarding the analytical modeling of the energy harvesting process, several research works have used different probability distributions to model energy sources. The work in [29] shows that no single probability distribution can fit all the empirical datasets of solar energy. The work in [30] shows that piezoeletric energy can be modeled by the generalized Markovian model, while solar energy can be modeled by a stationary Markovian model. The work in [31] proposes an analytical model in which the energy harvested in a time slot (of several seconds or minutes) is a random variable D. For mathematical tractability, the authors assume that D takes one of M discrete values $[d_1, ..., d_M]$ with probability $[p_1, ..., p_M]$. In this work, we use the approach of [31] and consider a probability distribution of the discrete exponential families (e.g., binomial, geometric) to model the energy harvested in a time interval T_R.

3. SYSTEM MODEL

In this section, we present the system model, which encompasses the network topology and the data generation, as well as the process that end-devices follow to harvest, store and consume energy.

3.1. Network and Data Model

We consider a wireless network in star topology formed by one coordinator (or gateway) and n end-devices in the communication range of the coordinator, as shown in Figure 1. Each end-device is equipped with a radio-transceiver, a micro-controller, several sensors, an energy harvester and an energy storage device (ESD). As depicted in Figure 2, the coordinator gathers data (e.g., measurements) from the end-devices by initiating periodic Data Collection Rounds (DCR). Each DCR starts when the coordinator broadcasts a Request for Data (RFD) packet, once every T_R seconds. In the k-th DCR, each end-device has a number $l(k)$ of new data packets ready to be transmitted to the coordinator. The data process $l(k)$ can be modeled as a discrete random variable with probability mass function $p_j = \Pr\{l(k) = j\}$ with $j \in \{1, 2, ...\}$. The value of $l(k)$ is considered to be identically and independently distributed (i.i.d.) over all end-devices and DCRs. We assume that the data packets have a common and constant length.

At the beginning of the k-th DCR, an end-device enters into *active mode* to transmit data if the energy available in its ESD is above a predefined energy threshold. Otherwise, the end-device remains in *sleep mode* waiting for the next DCR. In the example of Figure 2, end-devices 1, 2, and 4 have enough energy to become active in the k-th DCR, while end-device 3 remains in sleep mode. In the $(k + 1)$-th DCR, end-devices 2, 3, and 4 become active, while end-device 1 remains in sleep mode.

In this section, we describe the MAC protocols considered: TDMA, EH-RDFSA [28], and the EH-DQ protocol proposed in this paper.

The k-th DCR is formed by a sequence of F_k frames where each end-device in active mode transmits data to the coordinator according to the rules of the adopted MAC protocol. The coordinator broadcasts a feedback

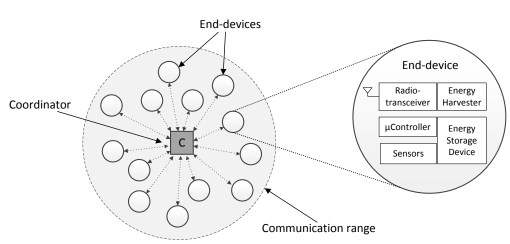


Figure 1. Wireless network with energy harvesting capabilities in star topology.

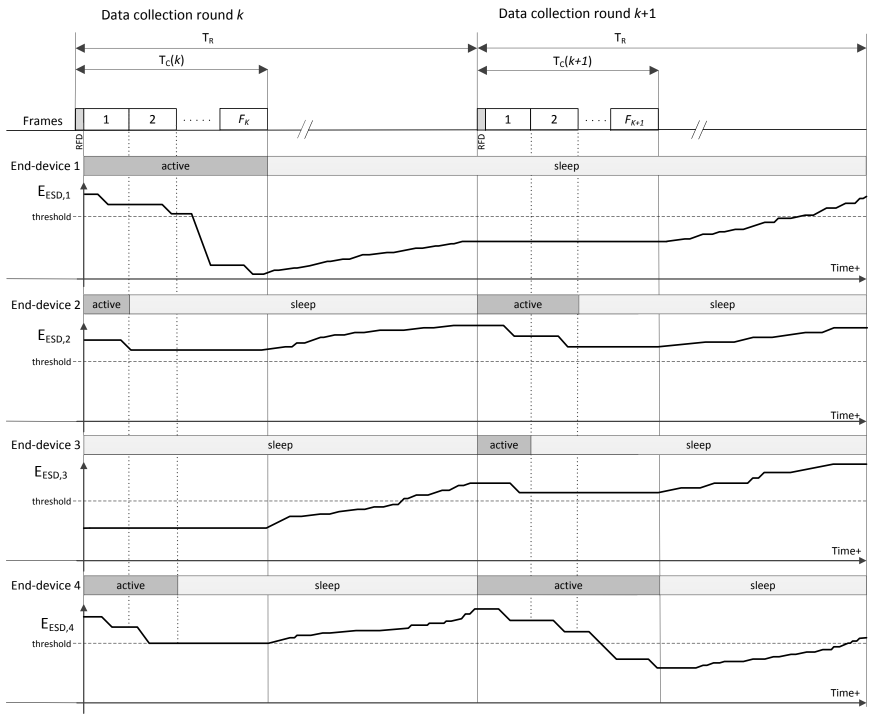


Figure 2. Sequence of data collection rounds with end-devices equipped with energy harvesters.

packet (FBP) or beacon at the end of each frame to enable the synchronization of the active end-devices and to inform about the number of slots of the next frame depending on the MAC layer. An active end-device attempts to transmit its $l(k)$ data packets, one-by-one sequentially in time, as long as it has enough energy in its ESD. During the k-th DCR, an end-device enters into *sleep mode* when it either succeeds in transmitting the last of the $l(k)$ data packets or falls in energy shortage. Therefore, the k-th DCR finishes when all the active end-devices have entered into sleep mode. We assume that the duration $T_C(k)$ of the k-th DCR is much shorter than the time T_R between any two consecutive DCRs (i.e., $T_C(k) \ll T_R$ for all k) to ensure that successive DCRs do not overlap.

We consider that if an end-device fails to transmit one or more data packets due to energy shortage in any given DCR, those data packets are discarded and they are not transmitted in subsequent DCRs. This scenario represents very well applications where a number of sensors is transmitting data and some packets can be lost or not transmitted without seriously affecting the application. Just as an example, a meter reader in a smart grid may fail to transmit one reading of the energy consumption, even though the next reading will implicitly include this information. Therefore, no re-transmission of a failed data packet would be needed.

In order to focus the study on the performance of the MAC layer, we assume that all packets are always transmitted without errors induced by the wireless channel. We assume that there is no capture effect, i.e., when two or more data packets collide, none of the packets involved in the collision can be decoded by the coordinator. The inclusion of transmission errors and capture effect constitutes part of our future work.

3.2. Energy Storage Model

The amount of energy stored in the ESD of an end-device can be modeled as a random variable which depends on the harvested energy and the energy consumed by the end-device throughout the DCRs. The energy stored in the ESD of the i-th end-device is denoted by $E_{ESD,i} \in \{0, 1\delta, 2\delta, ..., N\delta\}$, where δ [Joule] is referred to as *energy unit*, and N is the normalized capacity of the ESD. The end-device enters into active mode if the energy in its ESD at the beginning of the k-th DCR, denoted by $E_{ESD,i}(k)$, is above a certain energy threshold $E_{thr} = \varepsilon_{thr}\delta$, with $\varepsilon_{thr} \in \{0, 1, 2, ..., N-1\}$. The probability that an end-device can take part in the k-th DCR is called activation probability, denoted by $p_{active}(k)$, which can be expressed as

$$p_{active}(k) = \Pr\{E_{ESD,i}(k) > E_{thr}\}\,. \qquad (1)$$

The energy threshold ε_{th} must be selected so as to maximize performance.

3.3. Energy Harvesting Model

The energy harvester of the i-th end-device captures an amount of energy, denoted by $E_{H,i}(k)$, for the time interval T_R between any two consecutive DCRs $(k-1)$-th and k-th, which can be expressed as

$$E_{H,i}(k) = \int_{T_R} P_{H,i}(t)dt, \quad (2)$$

where $P_{H,i}(t)$ is the instantaneous electrical power delivered by the energy harvester.

The harvested energy $E_{H,i}(k)$ has been modeled as a discrete random variable with a probability mass function $q_j = \Pr\{E_{H,i}(k) = j\delta\}$ with $j \in \{0, 1, 2, ..., N_H\}$ energy units, which depends on the characteristics of the energy source and N_H is the maximum number of energy units that can be captured by the energy harvester. $E_{H,i}(k)$ is considered to be i.i.d. with regard to other end-devices and DCRs.

The *energy harvesting rate*, denoted by $\overline{E}_H$, is defined as the average harvested energy of an end-device during the time T_R between the beginning of two consecutive DCRs, which can be expressed as

$$\overline{E}_H = \mathbb{E}\left[E_{H,i}(k)\right]. \quad (3)$$

We assume that the dynamics of the energy harvesting process is slower than the contention process in a DCR. Therefore, we consider that the amount of energy that is harvested within the duration $T_c(k)$ of the k-th contention process is negligible with respect to $E_{H,i}(k)$, and it is not immediately available to be used during the contention process. Consequently, all the harvested energy $E_{H,i}(k)$ is ready to be used by an end-device at the beginning of the k-th DCR.

3.4. Energy Consumption Model

Regarding energy consumption, the end-devices can be in four different modes of operation: (i) transmitting a packet; (ii) receiving; (iii) standby, or (iv) sleeping. The associated power consumption are ρ_{tx}, ρ_{rx}, ρ_{stby}, or ρ_{sleep}, respectively. We assume that the energy required to switch between sleep and active modes (*i.e.*, transmitting, receiving) is negligible. In sleep mode, the radio interface is fully disabled, and thus, the end-devices consume the lowest power consumption.

In Section 5, we use the power consumption levels to formulate the analytical expressions required to compute the energy consumed by an end-device in a given frame by accounting for the energy used in all the communication phases (where the end-device is transmitting, in standby or receiving) depending on the type of frame.

4. MEDIUM ACCESS CONTROL PROTOCOLS

In this section, we describe the MAC protocols considered: TDMA, EH-RDFSA [28], and the EH-DQ protocol proposed in this paper.

4.1. Time Division Multiple Access (TDMA)

In TDMA, each frame is composed of a fixed number m_R of reserved slots that is equal to the number n of end-devices in the network. Each slot is allocated to one end-device. An end-device that becomes active in the k-th DCR will transmit its $l(k)$ data packets in its reserved slot in $l(k)$ successive frames (one packet per frame) as long as the end-device has enough energy. The coordinator responds with an acknowledgment packet (ACK) to each data packet decoded successfully within the same slot where the data packet has been transmitted. The coordinator broadcasts a feedback packet (FBP) or beacon at the end of each frame to enable the synchronization of the active end-devices.

4.2. Energy Harvesting-aware Reservation Dynamic FSA (EH-RDFSA)

In EH-RDFSA, each frame is composed of a variable number m_R of reserved slots and m_C contention slots. The number of reserved slots in the first frame is always 0. An end-device that becomes active in the k-th DCR randomly selects one of the contention slots in every frame to transmit the first of its data packets. A contention slot can be in one of three states: empty, if no packet has been transmitted in that slot; success, if only one packet has been transmitted; or collision, if more than one end-device has transmitted in that slot. When an end-device succeeds in transmitting the first packet in a given frame, one reserved slot will be allocated to the end-device for subsequent frames. The coordinator informs the end-device about the specific reserved slot with an ACK transmitted within the same slot where the data packet was transmitted. Then, the end-device transmits its other $l(k) - 1$ data packets in the reserved slot (one packet per frame) as long as it has enough energy. The coordinator responds with an ACK to each data packet decoded successfully in each slot. A reserved slot is released either once an end-device has transmitted all its data packets or enters in energy shortage. The header of every data packet includes a flag that indicates whether it is the last packet of the sequence of $l(k)$ packets to be transmitted in this DCR, and one field which informs about the energy available in the end-device. This information is used by the coordinator to calculate the number m_R of reserved slots in the next frame. In order to minimize $T_C(k)$, the coordinator adjusts the value of m_C to be equal to the number of active end-devices that contend to transmit their first data packet. The coordinator broadcasts a FBP at the end

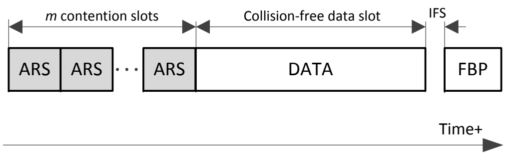


Figure 3. DQ frame structure.

of each frame to inform about the values of m_R and m_C for the next frame.

4.3. Energy Harvesting-aware Distributed Queuing (EH-DQ)

EH-DQ is the MAC protocol proposed in this work. In EH-DQ, the active end-devices request access to the channel in a short contention window at the beginning of each frame, thus confining collisions to a specific part of the frame. Collisions are resolved by using a tree-splitting algorithm [6] that organizes the end-devices into sub-groups to reduce the probability of collision per transmission attempt. When an end-device succeeds in transmitting its access request, it waits for its turn to transmit data in collision-free slots. Each frame of EH-DQ is divided in three parts as shown in Figure 3: *(i)* m contention slots devoted to the transmission of access request (ARS) packets, *(ii)* one collision-free slot for the transmission of a data packet, and *(iii)* a feedback packet (FBP). A guard time called Inter Frame Space (IFS) is left between reception and transmission modes to compensate propagation and processing delays and the time required to switch the radio transceivers between reception and transmission.

Every active end-device randomly selects one of the contention slots in every frame to transmit an ARS. Each ARS only contains one field (e.g., 1 byte) that indicates the number of data packets that must be transmitted by an end-device, i.e., the number l_R of collision-free slots to be reserved, which depends on the energy available in the end-device. Note that the ARS does not need to identify the end-device. Depending on whether the ARS collides or is successfully decoded by the coordinator, an end-device is queued into one of two logical and distributed queues:

1) The end-devices that have collided in a given contention slot when transmitting their ARS are queued into the *Collision Resolution Queue* (CRQ), sharing the same position in the queue. Note that after every frame, at most m new entries enter into the CRQ, being each one associated to each of the collisions that occurred in the last contention window. The length of the CRQ and the position of the end-devices in the CRQ is updated by executing the tree-splitting algorithm represented in Figure 4.a. Each node of the tree represents a frame of m contention slots (m=3 in the example), and the number in each contention slot denotes the number of end-devices that transmit an ARS in that slot. In every level of the tree, an end-device transmits its ARS in only one frame until it succeeds in one level or enters in energy shortage. The algorithm works as follows. At frame 1, all the active end-devices contend. If two or more end-devices collide in a slot, a new frame is assigned only to the end-devices that caused the collision in order to reattempt access, and they are queued into the CRQ. Therefore, if there are k slots with collision in one frame of level d, then k new frames are scheduled in level $d+1$, and k sub-groups of end-devices are queued into the CRQ. Once an end-device has entered in the CRQ, it will re-transmit its ARS in a given frame only if it has enough energy in its ESD and it occupies the first position in the CRQ; otherwise, the end-device enters in sleep mode and waits until it reaches the first position of the CRQ.

2) The end-devices that succeed in transmitting their ARS are queued into the *Data Transmission Queue* (DTQ). In principle, any queue discipline could be used to this end. For example, devices could enter into the DTQ following the same chronological order of the contention slots. Contrarily to the CRQ, in this case, every position of the DTQ is occupied by just one end-device. Indeed, an end-device occupies a number of positions in the DTQ that is equal to the number l_R of collision-free data slots reserved by the end-device for this particular DCR. When an end-device reaches the first position of the DTQ, it transmits its data packets in the collision-free slot of successive frames.

The CRQ and DTQ are represented at each end-device by 2 integer numbers per queue representing: *1)* the position of the end-device in the queue, and *2)* the total length of the queue. The length of the CRQ represents the number of sub-groups of end-devices waiting to re-transmit an ARS. The length of the DTQ represents the total number of collision-free slots reserved by the end-devices that have succeeded in transmitting their ARS and wait for their first collision-free slot.

The coordinator updates the length of the CRQ and DTQ at the end of each frame according to the following rules: *1)* the length of the CRQ is incremented by the number of contention slots with collision; *2)* if the CRQ was not empty in the previous frame, then its length is decremented by one; *3)* the length of the DTQ is incremented by the total number of collision-free slots reserved in each frame; and *4)* if the DTQ was not empty in the previous frame, then its length is decremented by one.

The coordinator broadcasts in every FBP: *(i)* the length of the CRQ (2 bytes); *(ii)* the length of the DTQ (2 bytes); and *(iii)* the state of the m contention slots (empty, success, or collision) and the number of collision-free slots reserved in every slot with one successful ARS (1 byte per contention slot). Using the information of the FBP, an end-device that transmitted an ARS can compute its position in the CRQ if it collides, or its position in the DTQ if it succeeds. The positions in the CRQ and DTQ are always decremented by one at the end of each frame. Therefore, the end-devices only receive the FBP in those frames where they transmit either an ARS or a data packet, and they enter into sleep mode in those frames where they do not transmit either ARS or data, in order to save energy.

Figure 4 shows an example of the operation of EH-DQ. The contents of the slots and the lengths of the CRQ and DTQ in every frame are shown in Figure 4.a. The contents of both queues are shown in Figure 4.b. At frame 1, all the end-devices (d1 to d6) transmit an ARS: d1, d2 and d3 collide in slot 1; d4 succeeds in slot 2; d5 and d6 collide in slot 3. Thus, d1, d2 and d3 enter in the first position of the CRQ; d4 enters in the first position of the DTQ reserving 1 collision-free slot; d5 and d6 enter in the second position of the CRQ. At frame 2, d4 transmits its data packet (because it occupies the first position of the DTQ), and d1, d2 and d3 transmit an ARS (because they occupy the first position of the CRQ): d1 and d2 collide and enter in the CRQ again; d3 succeeds and enters in the DTQ reserving 2 collision-free slots; d5 and d6 move to the first position of the CRQ. At frame 3, d5 and d6 transmit an ARS, collide, and enter in the second position of the CRQ again; d1 and d2 move to the first position of the CRQ; and d3 transmits its first data packet. At frame 4, d3 transmits its second data packet; d1 and d2 transmit an ARS, succeed, and enter in the DTQ reserving 1 collision-free slot each; and d5 and d6 move to the first position of the CRQ. The process continues until the end of the DCR.

5. ENERGY CONSUMPTION MODEL USING EH-DQ

In EH-DQ, every time that an end-device transmits an ARS in a certain frame of a DCR, it consumes a constant amount of energy, denoted by E_{ARS} [Joule], which accounts for the energy used in the following communication phases: *(i)* the end-device transmits the ARS in 1 contention slot, *(ii)* it remains in standby mode in the other $m-1$ contention slots and in the collision-free slot, and *(iii)* it receives the FBP. Then, E_{ARS} can be formulated as

$$E_{ARS} = \rho_{tx} T_{ARS} + (m-1)\,\rho_{stby} T_{ARS} + \qquad (4)$$

$$\rho_{stby} T_{data} + \rho_{rx} T_{FBP}, \qquad (5)$$

where T_{ARS}, T_{data} and T_{FBP} are the duration of a contention slot, a collision-free slot, and the time of transmission of a FBP, respectively; and ρ_{tx}, ρ_{rx} and ρ_{stby} are the power consumption in transmission, reception and standby mode, respectively.

Every time that an end-device transmits one data packet in a certain frame of a DCR, it consumes a constant amount of energy, denoted by E_{data} [Joule], which accounts for the energy used in the following communication phases: *(i)* the end-device remains in standby mode in m contention slots, *(ii)* it transmits data in the collision-free slot, and *(iii)* it receives the FBP. Then, E_{data} can be formulated as

$$E_{data} = m\rho_{stby} T_{ARS} + \rho_{tx} T_{data} + \rho_{rx} T_{FBP}. \qquad (6)$$

We assume that the energy consumed by an end-device in those frames where it is in sleep mode is negligible. For convenience, we normalize these energy consumption to $E_{ARS} = 1\delta$ and $E_{data} \approx K\delta$, where K is a positive integer number. Therefore, an end-device consumes 1 and K energy units δ when it transmits an ARS or a data packet in a frame, respectively.

6. PERFORMANCE METRICS

The *data delivery ratio* (DDR) is defined as the ratio between the number of data packets that are successfully transmitted to the coordinator in the k-th DCR, and the number of data packets ready to be transmitted at the beginning of the DCR. The DDR measures the ability of the MAC protocol to successfully deliver long data messages fragmented in small packets from the end-devices to the coordinator in every DCR without depleting their ESD.

The *time efficiency* is defined as the ratio between the duration of all the data packets successfully transmitted to the coordinator in the k-th DCR, and the time $T_C(k)$ required to complete the DCR. This value measures the probability that one slot allocated by the MAC layer during a DCR is successfully used. Therefore, the time efficiency is an indicator of the *data collection rate*, which can be obtained by dividing the time efficiency by the duration of a slot.

Due to the fluctuations of the harvested energy, the limited capacity of the ESDs, and collisions, both DDR and time efficiency may be lower than 1. Since the use of energy harvesters potentially allows for perpetual operation, it is interesting to analyze the performance metrics when the system is in *steady-state*, i.e., for a DCR with large index k.

7. ANALYSIS OF PERFORMANCE METRICS

In order to derive an analytic model to compute the DDR and the time efficiency in steady-state for EH-DQ, we need to evaluate the steady-state probability distribution of the energy available in the ESDs at the beginning of a DCR, which depends on the energy harvesting process, the random slot selection in every frame, the tree splitting process, and the number of data packets transmitted by each end-device in previous DCRs. Given that the number of end-devices that contend in every frame depends on the energy available in the

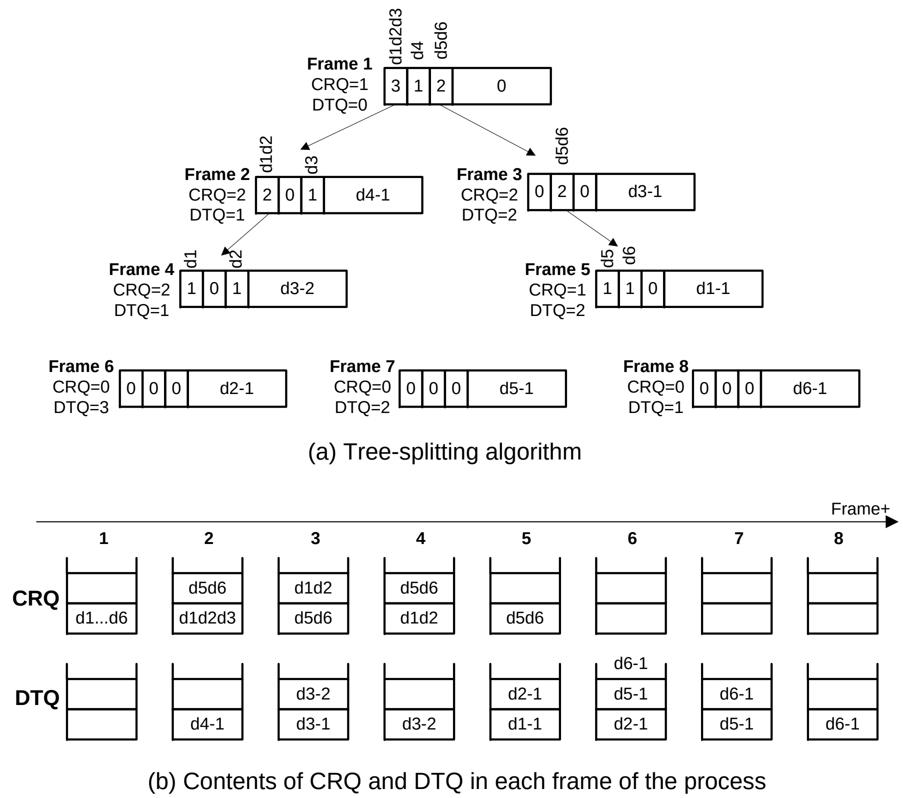


Figure 4. Example of EH-DQ with 6 end-devices (d1 to d6) and 3 contention slots: (a) tree-splitting algorithm, and (b) contents of the CRQ and DTQ in each frame.

ESD at each end-device, deriving the exact steady-state probability distribution is not an easy task. However, if we adjust the value of the energy threshold ε_{th} (1) to guarantee that all the end-devices that become active (n_1) in a DCR will have enough energy to contend in a certain number of levels, assuming that the number of end-devices that fall in energy shortage in a DCR is negligible, we can consider that the probability that an end-device succeeds in transmitting an ARS packet in one frame of any level of the contention tree basically depends on the value of n_1, the number of slots per frame, and the level number where the device contents. Consequently, we can evaluate the steady-state probability distribution of the energy available in the ESDs by analyzing the evolution of the energy of a single ESD, which is an approximation that neglects the interactions among the ESDs of different end-devices.

In this section, we derive an analytic model for EH-DQ to compute the DDR and the time efficiency in steady-state. To this end, in Section 7.1 we first propose a discrete-time Markov chain model to analyze the evolution of the energy available in the ESD of a given end-device. In Section 7.2, we derive the probability that an end-device succeeds in transmitting an ARS packet in one frame of a DCR. Finally, we derive in Section 7.3 the steady-state probability distribution of the energy available in the ESDs at the beginning of a DCR, we formulate the DDR in Section 7.4 and the time efficiency in Section 7.5.

7.1. Markov Chain Model

The evolution of the energy available in the ESD of an end-device can be modeled with the discrete-time Markov chain shown in Figure 5. Each state in the chain is defined by $\{e(t), d(t)\}$, where $e(t) \in \{0, 1, ..., N\}$ is a stochastic process that represents the number of energy units δ available in the ESD at time t; and $d(t) \in \{0, 1, ..., N-K\}$ is a stochastic process that represents that either an end-device is in sleep mode when $d(t) = 0$, or the level number in the contention tree where an end-device transmits an ARS when $d(t) \in \{1, ..., N-K\}$. Recall that in every level of the tree, an end-device transmits an ARS in only one frame. Note that the state transitions in the Markov chain do not occur at fixed time intervals.

The Markov chain is characterized by a transition matrix $\mathrm{P} = [p_{ij}]$, where each element p_{ij} is the one-step transition probability defined as

$$p_{ij} = \Pr\left\{e(t+1) = e_j, d(t+1) = d_j | e(t) = e_i, d(t) = d_i\right\} \ . \tag{7}$$

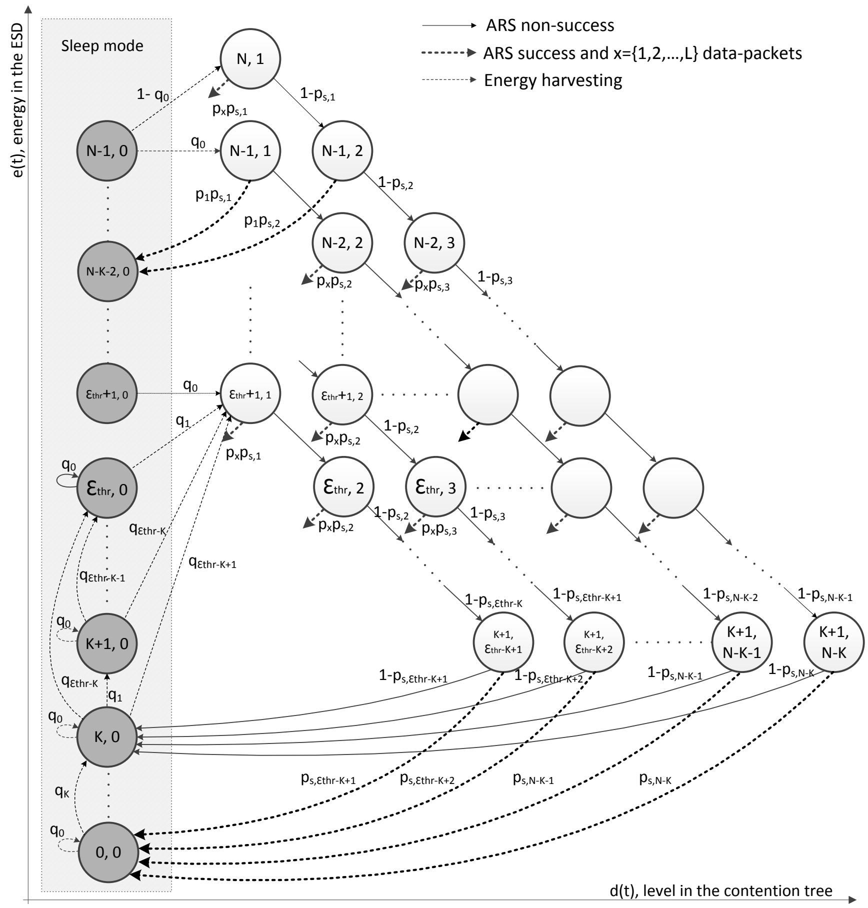


Figure 5. Generalized state transition diagram of the Markov chain that models the evolution of the energy available in an ESD using EH-DQ.

An end-device that has successfully transmitted all its data packets or entered in energy shortage in a DCR, remains in sleep mode (i.e., in a state with $d_i = 0$) until the next DCR starts. At the beginning of a DCR, the number ε_H of energy units harvested in the last T_R interval is added to the energy in the ESD, i.e., $e_j = e_i + \varepsilon_H$. Then, if the number of energy units available in the ESD is above the threshold ε_{th}, i.e., $e_j \in \{\varepsilon_{th} + 1, ..., N\}$, the state of the end-device changes from sleep $(e_i, 0)$ to active mode $(e_j, 1)$. Otherwise, if the number e_j of energy units in the ESD is below or equal to ε_{th}, the end-device makes a transition from state $(e_i, 0)$ to state $(e_j, 0)$ and remains in sleep mode. The transition probability from state $(e_i, 0)$ to any state (e_j, d_j) at the beginning of a DCR can be expressed as

where q_{ε_H} is the probability that an end-device harvests a number ε_H of energy units, being $\varepsilon_H = e_j - e_i$ with $e_i \leq e_j$.

Once an end-device becomes active at the beginning of a DCR, it will transmit an ARS packet in one frame of every successive level of the contention tree until either it succeeds or its ESD falls below $(1 + K)$ energy units. Recall that an end-device consumes 1 energy unit when transmits an ARS, and K energy units when transmits a data packet.

If the end-device does not succeed in transmitting an ARS in one frame of level $d_i \in \{1, ..., N - K\}$, it can make two possible transitions: *(i)* to state $(e_i - 1, d_i + 1)$, if the end-device has enough energy to re-transmit an ARS in the next level and to transmit one or more data packets, i.e., $e_i \in \{2 + K, ..., N\}$; or *(ii)* to state $(K, 0)$, if the end-device has not enough energy to re-transmit an ARS and a data packet, i.e., $e_i = 1 + K$.

Once an end-device succeeds in transmitting an ARS in one frame of level d_i, which happens with probability

$$p_{ij} = \begin{cases} q_{\varepsilon_H}, & if\,(e_i \le e_j)\,and\,(e_j \le \varepsilon_{th})\,and\,(d_j = 0) \\ q_{\varepsilon_H}, & if\,(e_i \le e_j)\,and\,(\varepsilon_{th} < e_j < N)\,and\,(d_j = 1) \\ 1 - \sum\limits_{k=0}^{N-1-e_i} q_k, & if\,(e_i < e_j)\,and\,(e_j = N)\,and\,(d_j = 1) \\ 0, & otherwise \end{cases}, \tag{8}$$

$p_{s,d}$ with $d = d_i$ (derived in Section 7.2), the end-device will transmit a number $l \in \{1, 2, ..., l_{max}\}$ of data packets in the collision-free slot of subsequent frames. The maximum value l_{max} is limited by the energy available in the ESD, i.e., $l_{max} = \left\lfloor \frac{e_i - 1}{K} \right\rfloor$. Thus, the end-device makes the following transitions from state (e_i, d_i): *(i)* to states $(e_i - 1 - lK, 0)$ with probability $p_{s,d} \cdot p_l$ for $l \in \{1, 2, ..., l_{max} - 1\}$, or *(ii)* to state $(e_i - 1 - l_{max}K, 0)$ with probability $p_{s,d} \cdot \left(1 - \sum\limits_{l=1}^{l_{max}-1} p_l\right)$ for $l \in \{l_{max}, ..., L\}$, where p_l is the probability that an end-device has a number $l \in \{1, 2, ..., L\}$ of data packets ready to transmit at the beginning of a DCR. Consequently, the transition probability from state (e_i, d_i) to state (e_j, d_j) with $d_i \in \{1, 2, ..., N - K\}$ can be formulated as in Equation (9).

7.2. Probability of Success in one Frame

The probability that an end-device succeeds in transmitting an ARS packet in one frame of level $d \in \{1, 2, ..., N - K\}$, denoted by $p_{s,d}$, can be expressed as

$$p_{s,d} = \left(1 - \frac{1}{m}\right)^{n_d - 1}, \tag{10}$$

where m is the number slots per frame and n_d is the number of end-devices which contend in one frame of level d.

In the first frame of a steady-state DCR, i.e., in level $d = 1$, the number n_1 of end-devices that contend is equal to the average number of end-devices that become active, which can be expressed as $n_1 = n \cdot p_{active}^{SS}$, where n is the total number of end-devices and p_{active}^{SS} is the activation probability in steady-state, i.e., for large index k of DCR, defined as

$$p_{active}^{SS} = \lim_{k \to \infty} p_{active}(k). \tag{11}$$

We can assume that all the end-devices that become active in a steady-state DCR will have enough energy to contend until they succeed in transmitting an ARS. Note that this can be guaranteed by properly adjusting the value of the threshold ε_{th}. Under this assumption, the value of n_d for $d > 1$ can be derived as follows. First, the probability that k of n_d end-devices transmit in the same slot of a frame, denoted by $p_s(k)$, can be calculated as

$$p_s(k) = \binom{n_d}{k} \left(\frac{1}{m}\right)^k \left(1 - \frac{1}{m}\right)^{n_d - k}, \tag{12}$$

and the average number of empty, success, and collision slots in that frame can be calculated as $S_d^E = m \cdot p_s(0)$, $S_d^S = m \cdot p_s(1)$, and $S_d^C = m - S_d^E - S_d^S$, respectively.

As described in Section 4.3, if there are S_d^C slots with collision in one frame of level d, then $F_{d+1} = S_d^C$ new frames are scheduled in level $d + 1$, where each new frame in level $d + 1$ is assigned only to the sub-group of end-devices that caused a collision in the same specific slot of level d. The average number of end-devices that succeed in one frame of level d, denoted by n_d^S, is equal to the average number S_d^S of slots with success. Therefore, the average number of end-devices that collide in one frame of level d, denoted by n_d^C, can be calculated as $n_d^C = n_d - S_d^S$. Since we assume that the n_d^C end-devices have enough energy, they will contend again in F_{d+1} new frames of level $d + 1$. Then, the average number of end-devices that contend in one frame of level $d + 1$ can be calculated as

The probability that an end-device succeeds in transmitting an ARS in one frame of every level of the contention tree (10) is represented in Figure 6a. It has been evaluated with $m \in \{3, 10\}$, $n \in \{10 \cdot m, 100 \cdot m\}$, $p_{active}^{SS} = 1$, and considering that all the end-devices that become active in a DCR have enough energy to contend until they succeed in transmitting their ARS packet in the DCR regardless of the energy harvesting rate and the capacity of the ESDs. Results show a tight match between analytic and simulated results. As it could be expected, the value of $p_{s,d}$ is close to 0 for low values of d, especially when the number m of slots is low and the number n of end-devices is high.

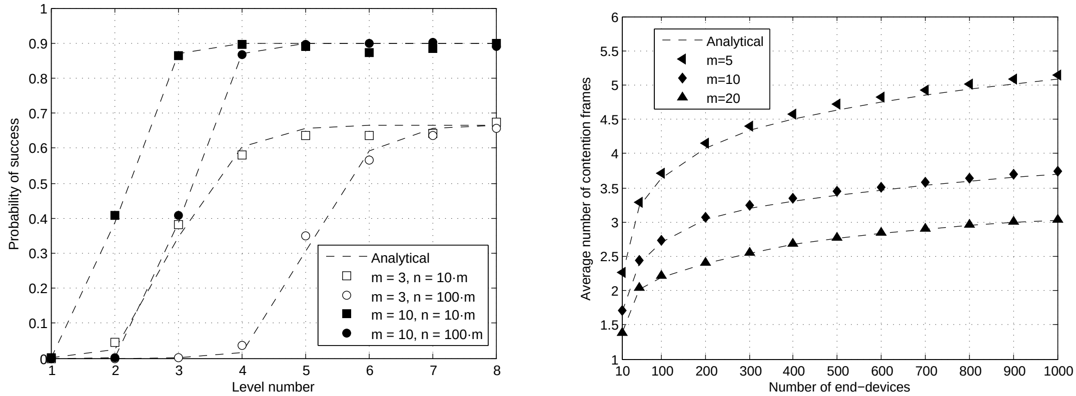
In order to set an appropriate value for the threshold ε_{th} which minimizes the probability that an end-device enters in energy shortage before it succeeds in transmitting an ARS, it is necessary to calculate the average number of frames where an end-device has to contend until it succeeds, denoted by $\mathbb{E}[d]$, which can be expressed as

$$\mathbb{E}[d] = \sum_{d=1}^{\infty} d \cdot p_{s,d} \cdot \prod_{i=1}^{d-1} (1 - p_{s,i}). \tag{14}$$

The value of $\mathbb{E}[d]$ is represented in Figure 6b as a function of the number of end-devices. It has been evaluated by considering $m \in \{5, 10, 20\}$. As it could be expected, the value of $\mathbb{E}[d]$ increases with n for a given value of m. The energy threshold needs to be adjusted as $\varepsilon_{th} \ge \mathbb{E}[d]$ depending on the values of m

$$p_{ij} = \begin{cases} (1-p_{s,d}), & if\,(e_i \geq 2+K)\,and\,(e_j = e_i - 1)\,and\,(d_j = d_i + 1) \\ (1-p_{s,d}), & if\,(e_i = 1+K)\,and\,(e_j = K)\,and\,(d_j = 0) \\ p_{s,d}\cdot p_l, & if\,(1 \leq l < l_{max})\,and\,(e_j = e_i - 1 - Kl)\,and\,(d_j = 0) \\ p_{s,d}\cdot\left(1-\sum\limits_{l=1}^{l_{max}-1} p_l\right), & if\,(l \geq l_{max})\,and\,(e_j = e_i - 1 - Kl_{max})\,and\,(d_j = 0) \\ 0, & otherwise \end{cases}. \quad (9)$$

$$n_{d+1} = \frac{n_d - S_d^S}{F_{d+1}} = \frac{n_d - n_d\left(1-\frac{1}{m}\right)^{n_d-1}}{m - m\left(1-\frac{1}{m}\right)^{n_d} - n_d\left(1-\frac{1}{m}\right)^{n_d-1}}. \quad (13)$$



(a) Probability of success

(b) Average number of contention levels

Figure 6. Probability that an end-device succeeds in transmitting an ARS in one frame of every level of the tree. Average number of levels where an end-device contends until it succeeds in transmitting an ARS in a DCR.

and n. For example, for a network of 1000 end-devices, the minimum energy threshold must be close to 5, 4, or 3 energy units when m is 5, 10, or 20 slots, respectively.

7.3. Steady-State Probability Distributions

As it can be observed in Figure 5, when $p_d > 0$ for $d \in \{1, 2, ..., N-K\}$, $q_0 > 0$ and $q_1 > 0$, the Markov chain is aperiodic and any state of the Markov chain can be reached from any other state with non-zero probability, and therefore the Markov chain is irreducible [32].

Since the Markov chain is irreducible and aperiodic, and thus ergodic, it admits a unique steady-state probability distribution [32], denoted by $\pi = [\pi_{e,d}]$, which can be expressed as

$$\pi_{e,d} = \lim_{t\to\infty} \Pr\{e(t) = e, d(t) = d\}, \quad (15)$$

and satisfies that

$$\left(\mathrm{P}' - \mathrm{I}\right)\pi' = 0, \quad (16)$$

where P is the transition matrix and I is the identity matrix. Equation (16) can be solved for π by calculating the eigenvector of P' that corresponds to an eigenvalue equal to 1. The steady-state probability distribution π is equal to the eigenvector with its elements normalized to sum one.

Recall that the transition matrix P depends on $p_{s,d}$ (10), which also depends on p_{active}^{SS} (11). On the other hand, p_{active}^{SS} can be expressed from the steady-state probability distribution of the energy available in the ESD at the beginning of a DCR, denoted by $\pi^B = \left[\pi_{e,d}^B\right]$, as follows

$$p_{active}^{SS} = \pi_{\varepsilon_{thr}+1,1}^B + ... + \pi_{N,1}^B = \sum_{e=\varepsilon_{thr}+1}^{N} \pi_{e,1}^B. \quad (17)$$

Note that all the values of π^B are zero for $d \in \{2, ..., N-K\}$. This is due to the fact that at the beginning of a DCR an end-device can only reach either states $(e, 0)$ with $e \in \{0, 1, ..., \varepsilon_{th}\}$ or $(e, 1)$ with $e \in \{\varepsilon_{th}+1, ..., N\}$. Since an end-device is in sleep mode before a DCR starts, π^B can be expressed as

$$\pi^B = \pi^S \mathrm{P}, \quad (18)$$

where $\pi^S = \left[\pi_{e,d}^S\right]$ is the steady-state probability distribution conditioned on being in sleep mode, which is calculated as

$$\pi^S_{e,d} = \begin{cases} \frac{\pi_{e,0}}{\sum\limits_{i=0}^{N-1} \pi_{i,0}}, & if\,(d=0) \\ 0, & if\,(1 \leq d \leq N-K) \end{cases}. \tag{19}$$

Finally, we compute the steady-state probability distributions as follows. Firstly, we build the transition matrix P by setting the steady-state activation probability to a test value of 0, i.e., $p^{SS}_{active-test} = 0$. Secondly, we solve equations (16), (19), and (18) to calculate π, π^S, and π^B, respectively. Thirdly, we compute the analytic value of p^{SS}_{active} (17) by using π^B. And finally, we check the relative error between the test and analytic values of the activation probability. These steps are repeated iteratively by increasing $p^{SS}_{active-test}$ until the error is below 0.1%, which indicates that it satisfies (16), (19) and (18), and the results obtained for π, π^S, and π^B are correct.

7.4. Data Delivery Ratio

Once the steady-state probability distribution π^B of the energy available in the ESD at the beginning of a DCR is computed, we can formulate the expression to calculate the *data delivery ratio* in steady-state for EH-DQ as follows

$$\mathrm{DDR} = \frac{\mathbb{E}[N_S]}{\mathbb{E}[N_R]} = \frac{\sum\limits_{l=1}^{L} \mathbb{E}[N_d(l)] \cdot p_l}{\sum\limits_{l=1}^{L} l \cdot p_l}, \tag{20}$$

where $\mathbb{E}[N_S]$ is the average number of data packets that are successfully transmitted to the coordinator in a DCR, $\mathbb{E}[N_R]$ is the average number of packets ready to be transmitted at the beginning of the DCR, and $\mathbb{E}[N_d(l)]$ is the average number of packets successfully transmitted by an end-device when it has $l \in \{1, 2, ..., L\}$ packets ready at the beginning of the DCR, which can be expressed as

Recall that an end-device which enters in active mode re-transmits an ARS packet in subsequent frames until it is successfully decoded by the coordinator. Then, the end-device transmits a number l_R of data packets which depends on the number l of packets ready, the amount of energy e available at the beginning of the DCR, and the level number d where the ARS succeeds, i.e., $l_R = \min\left(l, \left\lfloor \frac{e-d}{K} \right\rfloor\right)$.

7.5. Time Efficiency

The *time efficiency* for EH-DQ, denoted by $\jmath_t$, can be formulated as where $\mathbb{E}[N_E]$ is the average number of frames with the collision-free slot empty, i.e., frames which do not contain data. Since every active end-device transmits a number $l_R \geq 1$ of data packets, we can assume that once an end-device has first succeeded in transmitting an ARS in a given frame, every frame until the end of the DCR contains data. Therefore, $\mathbb{E}[N_E]$ can be approximated as the average number of frames where an end-device contends until it succeeds, i.e., $\mathbb{E}[N_E] \approx \mathbb{E}[d]$ (14). Since $\mathbb{E}[d] \ll \mathbb{E}[N_S]$, the expression of $\jmath_t$ can be approximated as

$$\jmath_t \simeq \frac{T_{data}}{mT_{ARS} + T_{data} + T_{FBP}}, \tag{23}$$

where recall that T_{ARS}, T_{data} and T_{FBP} are the duration of a contention slot, a collision-free slot, and the time of transmission of a FBP, respectively.

8. PERFORMANCE EVALUATION

In this section, we evaluate the performance of EH-DQ, in terms of the DDR and the time efficiency, and compare it with the performance of TDMA and EH-RDFSA. While in the theoretical model of EH-DQ the steady-state probability distribution of the energy in the ESDs is calculated by analyzing the evolution of the energy of a single ESD, which is an approximation of the actual model, the simulation does not neglect the interactions among the ESDs of different end-devices.

In the following sections, we first describe the considered scenario, and then discuss the numerical results to show how the performance is influenced by the energy threshold, the number of contention slots, and the energy harvesting rate.

8.1. Scenario

We consider a wireless network formed by 1 coordinator and a large number $n = 1000$ of end-devices. Each end-device periodically acquires measurements from a set of sensors and generates $L = 5$ data packets to be transmitted to the coordinator in every DCR. Each data packet has a payload of 114 bytes. At the end of each frame of EH-DQ, the coordinator broadcasts a FBP with a payload of 24 bytes that informs about the length of the CRQ and the DTQ, the status of the contention slots, and the number of collision-free slots reserved in every contention slot. All the packets are composed of a physical layer preamble, a MAC header, a payload and a cyclic redundancy code (CRC) of 2 bytes.

The system parameters used to evaluate the performance are summarized in Table I. They have been selected according to the IEEE 802.15.4 standard [33] and from the specifications of the CC2520 radio transceiver [34]. The values of energy consumption have been computed by using the energy consumption model described in Section 5. In particular, the energy consumed by an end-device when transmits an ARS packet in one frame, $E_{ARS} = 1\delta = 143$ μJoule, has been calculated from Equation (4), and the energy consumed by an end-device when transmits a data packet in one frame, $E_{data} = 4\delta$, has been calculated from Equation (6).

$$\mathbb{E}\left[N_d(l)\right] = \sum_{d=1}^{N-K} \sum_{e=d+K}^{N} \pi_{e,1}^{B} \prod_{i=1}^{d-1} \left(1 - p_{s,i}\right) p_{s,d} \cdot \min\left(l, \left\lfloor \frac{e-d}{K} \right\rfloor\right) \ . \tag{21}$$

$$\mathrm{J}_t = \frac{\mathbb{E}\left[N_S\right] T_{data}}{\left(\mathbb{E}\left[N_E\right] + \mathbb{E}\left[N_S\right]\right)\left(mT_{ARS} + T_{data} + T_{FBP}\right)}, \tag{22}$$

Each end-device includes an energy harvester and an ESD with $N = 40$ energy units δ of capacity. We assume that the energy harvested by an end-device in a DCR follows a binomial distribution with probability mass function

$$q_j = \binom{N_H}{j} \left(\frac{\overline{E_H}}{N_H}\right)^j \left(1 - \frac{\overline{E_H}}{N_H}\right)^{N_H - j} \tag{24}$$

for $j \in \{0, 1, 2, ..., N_H\}$, where $N_H = 40$ is the maximum number of energy units that can be captured and $\overline{E_H} \in [0, ..., N_H]$ is the average number of energy units harvested per DCR, i.e., the energy harvesting rate.

Results for EH-DQ have been obtained analytically and through computer-based simulations using MATLAB*. The results of 1000 simulation samples have been averaged for each test case. The tight match between analytic and simulation results validate the accuracy of the analytic model proposed in Section 7.

In TDMA and EH-RDFSA, we consider that an end-device consumes $K = 4$ energy units δ when transmits a data packet in one frame. We consider an ideal EH-RDFSA where the number of contenders per frame is perfectly estimated and the number of contention slots is adjusted in every frame to be equal to the number of end-devices that contend in order to transmit their first data packet, i.e., $\rho = 1$. Results for TDMA and EH-RDFSA have been obtained through computer-based simulations.

*The MATLAB simulation source code is available from the authors upon request. Please contact the corresponding author in case you are interested in obtaining it to reproduce the paper results and/or extend the simulations.

Table I. System parameters.

Parameter	Value	Parameter	Value
MAC header	8 bytes	Data-rate	250 kbps
Data payload	114 bytes	T_{data}	4.1 ms
ARS payload	1 byte	T_{ARS}	512 μs
FBP payload	24 bytes	T_{FBP}	1.2 ms
ρ_{tx}	100.8 mW	ρ_{stby}	525 μW
ρ_{rx}	66.9 mW	ρ_{sleep}	90 nW
E_{ARS}	1δ	E_{data}	4δ

8.2. Energy Threshold

The DDR and the time efficiency for EH-DQ, EH-RDFSA and TDMA are represented in Figure 7a and Figure 7b, respectively, as a function of the energy threshold ε_{thr} by considering $\overline{E_H} \in \{10, 20\}$ and $m \in \{3, 10\}$, where m is the number of contention slots in one frame of EH-DQ. As it can be observed, the value of the DDR for EH-DQ, EH-RDFSA and TDMA increases with the energy harvesting rate. Indeed, the higher the number of energy units available in the ESDs, the higher the number of end-devices that become active in a DCR and the higher the number of possible packet transmissions.

Recall that an end-device becomes active at the beginning of a DCR if the energy available in its ESD is above ε_{thr} energy units, and in EH-DQ an end-device consumes 1 energy unit when it transmits an ARS, and $K = 4$ energy units when it transmits a data packet. As shown in Figure 6b, the average number of frames in which an end-device has to contend until succeeds in transmitting an ARS, $\mathbb{E}[d]$ (14), is close to 5, 4, or 3 frames when $n = 1000$ and m is 5, 10, or 20 slots, respectively. Therefore, an end-device consumes an average of $\mathbb{E}[d]$ energy units in the transmission of ARS packets. Consequently, an end-device will need at least an energy level of $\varepsilon_{thr} \simeq \mathbb{E}[d] + KL$ in its ESD at the beginning of a DCR in order to maximize the DDR. As it can be observed in Figure 7a, the optimum value of ε_{thr} that maximizes the DDR for EH-DQ is within 20-25 energy units when m is 3 or 10 slots, $n = 1000$ end-devices, and $L = 5$ data packets.

The DDR for EH-RDFSA increases with the energy threshold, but much more slightly than in EH-DQ. This is due to the fact that in EH-RDFSA, since we consider that the number of contention slots per frame is adjusted to be equal to the number of end-devices that contend in every frame, the probability that an end-device succeeds in a given frame of EH-RDFSA is approximately constant (≈ 0.36) for all the frames. However, as it can be observed in Figure 6a, in EH-DQ the probability that an end-device succeeds in transmitting an ARS is very low in the first levels of the contention tree and then it increases above 0.36 when the level number increases. For example, when $m = 10$ slots and $n = 10 \cdot m$ end-devices, the probability that an end-device succeeds in one frame of level 2 and 3 is 0.4 and 0.9, respectively.

While the DDR for EH-DQ and EH-RDFSA increases with the energy threshold until it reaches its maximum value, results show that the DDR for TDMA does not increase with the energy threshold. This is due to the fact that in TDMA there is no energy waste in collisions and thus the DDR for TDMA only depends on the energy harvesting rate.

Finally, as it can be observed in Figure 7a, the DDR for EH-DQ, EH-RDFSA and TDMA decays dramatically when the energy threshold increases above a certain value. Indeed, when the energy threshold is too high, the activation probability decreases, thus reducing the DDR.

As it can be observed in Figure 7b, the time efficiency for EH-DQ decreases as the number m of contention slots increases. This is due to the fact that once all the end-devices have succeeded in transmitting their ARS, the higher the number of contention slots per frame, the higher the overhead and the time wasted in the DCR. In addition, as it could be expected according to Equation (23), the time efficiency for EH-DQ is insensitive to the energy harvesting rate and the energy threshold.

In contrast, the time efficiency for TDMA decreases as the energy threshold increases. This is due to the fact that the higher the energy threshold, the lower the number of end-devices that become active in a DCR, the higher the number of empty slots, and thus the lower the time efficiency. The time efficiency for TDMA increases as the energy harvesting rate increases. Indeed, the higher the energy harvesting rate, the higher the energy available in the ESDs at the beginning of a DCR, and the higher the number of possible packet transmissions per end-device. Consequently, the probability that one slot in every frame of TDMA contains one successful data packet increases with the energy harvesting rate.

Similarly, the time efficiency using EH-RDFSA increases with the energy harvesting rate and the energy threshold. Indeed, the probability that one reserved slot in every frame of EH-RDFSA contains one successful data packet increases with the energy available in the ESDs at the beginning of a DCR.

8.3. Number of Contention Slots

The DDR and the time efficiency of EH-DQ are represented in Figure 8a and Figure 8b, respectively, as a function of the number m of contention slots (from 2 to 20 slots) by considering $\overline{E_H} \in \{5, 10, 20, 30\}$ and $\varepsilon_{thr} = 20$, which is a value of the energy threshold close to the one that maximizes the DDR and the time efficiency for EH-DQ, EH-RDFSA and TDMA, as it can be observed in Figure 7a and Figure 7b. Recall that in EH-RDFSA the number of contention slots per frame is adjusted to be equal to the number of contenders in every frame, and in TDMA the number of slots is equal to the total number of end-devices in the network.

Results show that the DDR for EH-DQ increases when the number of contention slots per frame increases. Indeed, the higher the number of contention slots, the lower the probability that an ARS collides in a given frame, and the lower the energy wasted in re-transmissions, thus increasing the DDR.

The DDR for EH-DQ and EH-RDFSA increases with the energy harvesting rate. Indeed, the higher the number of energy units available in the ESDs, the higher the number of end-devices that become active in a DCR and the higher the number of possible packet re-transmissions.

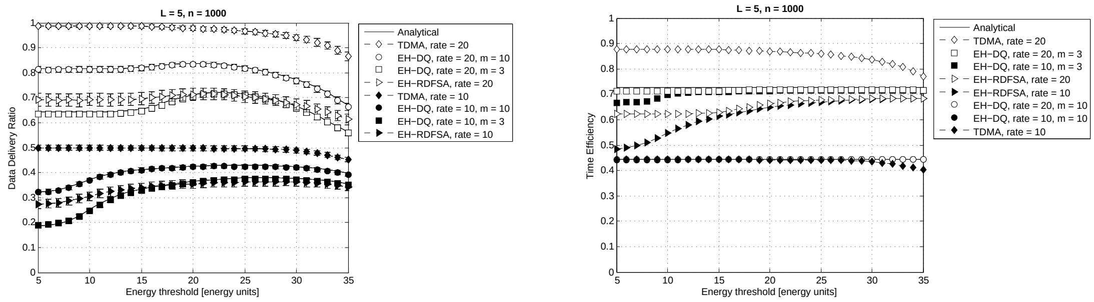
As it can be observed in Figure 8a, EH-DQ can outperform EH-RDFSA in terms of DDR, for any energy harvesting rate, if the number of slots per frame in EH-DQ is properly adjusted. For example, if $n = 1000$ and $\overline{E_H} \in \{5, 10, 20, 30\}$, then the value of m in EH-DQ must be equal or greater than 3 slots.

As it can be observed in Figure 8b, the time efficiency for EH-DQ is maximized for 2-3 contention slots, $\jmath_t \approx 0.80$, and it is degraded as the number of contention slots per frame increases. Indeed, the higher the number of contention slots, the higher the time wasted in every frame once all the end-devices have succeeded in transmitting their ARS, thus reducing the time efficiency. In addition, EH-DQ can outperform EH-RDFSA and TDMA in terms of time efficiency if the number of contention slots is low, and the time efficiency for EH-DQ is very similar for different energy harvesting rates.

There is a trade-off between DDR and time efficiency for EH-DQ. When the number of contention slots per frame increases, more end-devices can eventually succeed in transmitting ARS packets to the coordinator in a DCR, thus increasing the DDR, at the cost of reducing the time efficiency and the data collection rate. However, as it can be observed in Figure 8a, with low (e.g., 5) and high (e.g., 30) energy harvesting rates, EH-DQ can be configured with a very low number of contention slots (e.g., $m = 3$), at almost no cost in the DDR, and increase the time efficiency to a certain value close to the maximum. However, with intermediate energy harvesting rates (e.g., between 10 and 20 energy units), EH-DQ must be configured with a number of contention slots per frame which depends on the harvesting rate.

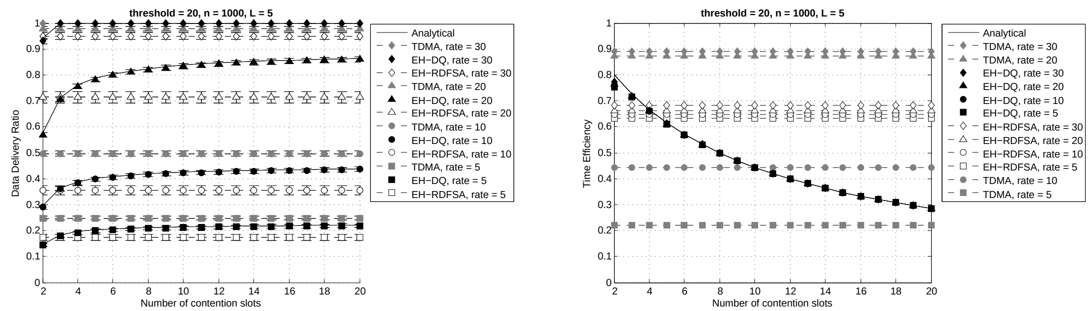
8.4. Energy Harvesting Rate

The DDR and the time efficiency for EH-DQ, EH-RDFSA and TDMA are represented in Figure 9b and Figure 9a, respectively, as a function of the energy harvesting rate $\overline{E_H}$ by considering $m \in \{3, 10\}$ and $\varepsilon_{thr} = 20$, which is a value of the energy threshold close to the one that maximizes the DDR and the time efficiency for EH-DQ, EH-RDFSA and TDMA, as it can be observed in Figure 7a and Figure 7b.



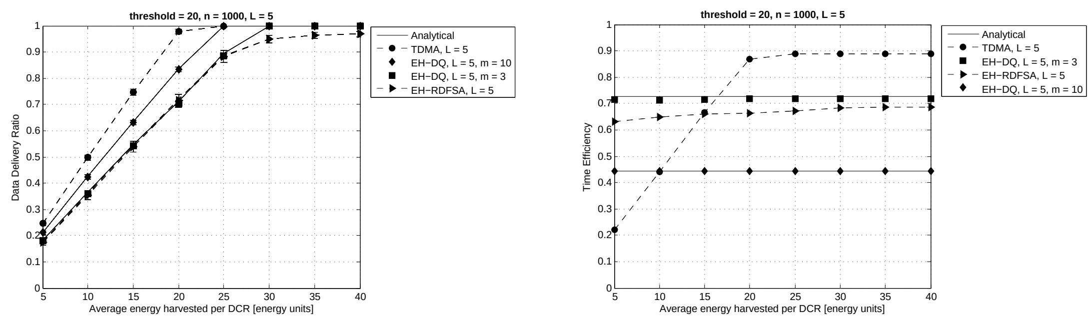
(a) Data Delivery Ratio **(b)** Time Efficiency

Figure 7. Data Delivery Ratio and Time Efficiency over the energy threshold using EH-DQ, EH-RDFSA and TDMA.



(a) Data Delivery Ratio **(b)** Time Efficiency

Figure 8. Data Delivery Ratio and Time Efficiency over over the number of contention slots per frame.



(a) Data Delivery Ratio **(b)** Time Efficiency

Figure 9. Data Delivery Ratio and Time Efficiency over the energy harvesting rate using EH-DQ, EH-RDFSA and TDMA.

The value of the DDR increases almost linearly with $\overline{E_H}$ for EH-DQ, EH-RDFSA, and TDMA. Indeed, the higher the energy available in an ESD at the beginning of a DCR, the higher the number of possible packet transmissions and thus the value of the DDR. As it could be expected, TDMA yields a value of DDR equal to 1 when $\overline{E_H} \geq 4L = 20$ energy units. Indeed, since there are no collisions in TDMA, its performance is only limited by the amount of harvested energy and the capacity of the ESD. Note that ideal TDMA could be considered as the upper bound for random access protocols in terms of absence of collisions. In EH-RDFSA, however, the value of the DDR is close to 1 when $\overline{E_H} > 5L$. In its turn, EH-DQ attains a value of the DDR equal to 1 when $\overline{E_H} \geq 5L = 25$ energy units for $m = 10$, and when $\overline{E_H} \geq 6L = 30$ for $m = 3$. Indeed, as the probability of collision is lower when the number of contention slots increases, the energy consumption due to re-transmissions of ARS packets is reduced, thus increasing the DDR. In addition, it can be observed that EH-DQ outperforms the DDR provided by EH-RDFSA. Indeed, as in EH-DQ the end-devices only contend to transmit short ARS packets, the collisions and the energy consumption due to re-transmissions are reduced with respect to EH-RDFSA, thus increasing the DDR. Furthermore, the tree splitting algorithm of

EH-DQ allows that the ARS packets can be eventually transmitted with a finite number of re-transmissions. Results show that EH-DQ with $m = 10$ requires lower energy harvesting rate than EH-RDFSA to get the same DDR. For example, while EH-RDFSA requires $\overline{E_H} = 30$ energy units to obtain DDR $= 0.95$ with $L = 5$, EH-DQ requires $\overline{E_H} = 23$, which means a reduction of 23% in energy harvesting rate. Consequently, EH-DQ allows reducing the total time between consecutive DCRs and thus increases the network throughput with respect to EH-RDFSA.

As it can be observed in Figure 9a, the time efficiency for TDMA increases with $\overline{E_H}$ and tends to its maximum value when $\overline{E_H} > 4L$. Indeed, the higher the energy harvesting rate, the higher the energy available in the ESDs at the beginning of a DCR, and the higher the number of possible packet transmissions per end-device. Consequently, the probability that one slot in every frame of TDMA contains one successful data packet increases with $\overline{E_H}$. In EH-RDFSA, the time efficiency increases slightly with $\overline{E_H}$. Contrarily, the time efficiency of EH-DQ is insensitive to the energy harvesting rate.

While the time efficiency in TDMA increases linearly up to 0.9, which indicates that every slot contains one successful data packet, the maximum time efficiency in EH-RDFSA is 0.7, and it is 0.72 and 0.45 in EH-DQ with $m = 3$ and $m = 10$ number of contention slots, respectively. Indeed, while in TDMA each end-device transmits in its reserved slot, in EH-RDFSA and EH-DQ the end-devices have to contend until they succeed in transmitting their first data packet and the ARS, respectively, with the consequent waste of time in contention slots.

When the energy harvesting rate is below a certain threshold, the time efficiency in EH-DQ is greater than in TDMA. As it can be observed in Figure 9a, EH-DQ outperforms TDMA when $\overline{E_H} < 3L$. Indeed, while the number of slots per frame in TDMA is constant, equal to the total number of end-devices in the network regardless of the number of active end-devices, every frame in EH-DQ contains a very short contention window and 1 collision-free slot reserved for a specific end-device, thus leading to higher time efficiency.

9. CONCLUSIONS

In this paper, we have proposed a new MAC protocol, named Energy Harvesting-aware Distributed Queuing Access (EH-DQ), for data collection networks where each end-device is equipped with an energy harvester and generates messages which are fragmented into small data packets to be transmitted to a gateway. We have considered the data delivery ratio (DDR) and the time efficiency as performance metrics. We have modeled the operation of EH-DQ with a discrete-time Markov chain to analyze the evolution of the energy availability and to calculate the performance metrics. We have compared the performance of EH-DQ with the EH-aware Reservation Dynamic Frame Slotted-ALOHA (EH-RDFSA) and the upper-bound performance of an ideal TDMA protocol. Results show that the DDR increases with the energy harvesting rate for all cases. EH-DQ and TDMA provide the maximum DDR $= 1$, and both outperform EH-RDFSA in terms of DDR and time efficiency. While the time efficiency of TDMA increases with the energy harvesting rate, the time efficiency of EH-DQ is insensitive to the harvesting rate. EH-DQ outperforms TDMA in terms of the time efficiency in a certain range of the energy harvesting rate which depends on the number of data packets to be transmitted by each end-device. Furthermore, while EH-RDFSA requires to estimate the number of contenders in each frame to adapt the frame length dynamically, EH-DQ uses a short and fixed frame length. In addition, while TDMA requires to update the knowledge of the network topology to maintain a collision-free schedule, EH-DQ does not require topology knowledge, thus reducing overhead and energy consumption. Taking that into account, we believe that EH-DQ is an interesting alternative for data collection scenarios with energy harvesting. Future work aims at including transmission errors and capture effect in the analysis presented in this paper.

ACKNOWLEDGMENTS

This work was supported by the Research Projects CellFive (TEC2014-60130-P), P2P-SmarTest (H2020-646469), by the Spanish Ministry of Economy and the FEDER regional development fund through the SINERGIA project (TEC2015-71303-R), and by the Generalitat de Catalunya under Grant 2014 SGR 1551.

REFERENCES

1. A. Bachir, M. Dohler, T. Watteyne, K. K. Leung, Mac essentials for wireless sensor networks, IEEE Communications Surveys Tutorials 12 (2) (2010) 222–248. doi:10.1109/SURV.2010.020510.00058.
2. G. Anastasi, M. Conti, M. Di Francesco, A. Passarella, Energy conservation in wireless sensor networks: A survey, Ad Hoc Networks 7 (3) (2009) 537–568. doi:10.1016/j.adhoc.2008.06.003. URL `http://dx.doi.org/10.1016/j.adhoc.2008.06.003`
3. B. Martinez, M. Montn, I. Vilajosana, J. D. Prades, The power of models: Modeling power consumption for iot devices, IEEE Sensors Journal 15 (10) (2015) 5777–5789.

4. S. Ulukus, A. Yener, E. Erkip, O. Simeone, M. Zorzi, P. Grover, K. Huang, Energy harvesting wireless communications: A review of recent advances, Selected Areas in Communications, IEEE Journal on 33 (3) (2015) 360–381. doi:10.1109/JSAC.2015.2391531.
5. B. Martinez, X. Vilajosana, F. Chraim, I. Vilajosana, K. S. J. Pister, When scavengers meet industrial wireless, IEEE Transactions on Industrial Electronics 62 (5) (2015) 2994–3003. doi:10.1109/TIE.2014.2362891.
6. J. I. Capetanakis, Tree Algorithms for Packet Broadcast Channels, IEEE Transactions on Information Theory 25 (5) (1979) 505–515.
7. F. Vazquez-Gallego, J. Alonso-Zarate, P. Tuset-Peiro, L. Alonso, Energy performance of distributed queuing access in machine-to-machine networks with idle-to-saturation transitions, in: 2013 IEEE Globecom Workshops (GC Wkshps), 2013, pp. 867–872. doi:10.1109/GLOCOMW.2013.6825098.
8. F. Vazquez-Gallego, L. Alonso, J. Alonso-Zarate, Energy Harvesting-aware Distributed Queuing Access for Wireless Machine-to-Machine Networks, in: in proc. of the IEEE GLOBECOM 2016, Washington DC, USA, 2016.
9. W. Xu, G. Campbell, A Near Perfect Stable Random Access Protocol for a Broadcast Channel, in: IEEE International Conference on Communications (ICC), Vol. 1, 1992, pp. 370–374. doi:10.1109/ICC.1992.268230.
10. W. Chien-Ting, G. Campbell, Extended DQRAP (XDQRAP). A cable TV protocol functioning as a distributed switch, in: International Workshop on Community Networking Integrated Multimedia Services to the Home, 1994, pp. 191–198. doi:10.1109/CN.1994.337349.
11. L. Harn-Jier, G. Campbell, PDQRAP-prioritized distributed queueing random access protocol, in: IEEE Conference on Local Computer Networks, 1994, pp. 82–91. doi:10.1109/LCN.1994.386612.
12. W. Chien-Ting, G. Campbell, Interleaved DQRAP with Global TQ, Dept. of Comp. Sci., Illinois Inst. of Technology, 1995.
13. L. Alonso, R. Agusti, O. Sallent, A near-optimum MAC protocol based on the distributed queueing random access protocol (DQRAP) for a CDMA mobile communication system, IEEE Journal on Selected Areas in Communications 18 (9) (2000) 1701–1718. doi:10.1109/49.872957.
14. J. Alonso-Zarate, C. Verikoukis, E. Kartsakli, A. Cateura, L. Alonso, A Near-Optimum Cross-Layered Distributed Queuing Protocol for Wireless LAN, IEEE Wireless Communications 15 (1) (2008) 48–55. doi:10.1109/MWC.2008.4454704.
15. J. Alonso-Zarate, L. Alonso, C. Skianis, C. Verikoukis, Analysis of a Distributed Queuing Medium Access Control Protocol for Cooperative ARQ, in: IEEE GLOBECOM, 2010, pp. 1–5. doi:10.1109/GLOCOM.2010.5683302.
16. J. Alonso-Zarate, D. Gregoratti, P. Giotis, C. Verikoukis, L. Alonso, Medium access control priority mechanism for a DQMAN-based wireless network, IEEE Communications Letters 13 (7) (2009) 495–497. doi:10.1109/LCOMM.2009.081502.
17. B. Otal, L. Alonso, C. Verikoukis, Energy-Efficiency Analysis of a Distributed Queuing Medium Access Control Protocol for Biomedical Wireless Sensor Networks in Saturation Conditions, Sensors 11 (2) (2011) 1277–1296. doi:10.3390/s110201277.
URL `http://www.mdpi.com/1424-8220/11/2/1277`
18. F. Vazquez-Gallego, J. Alonso-Zarate, P. Tuset-Peiro, L. Alonso, Energy Analysis of a Contention Tree-based Access Protocol for Machine-to-Machine Networks with Idle-to-Saturation Traffic Transitions, in: IEEE International Conference on Communications (ICC), 2014.
19. P. Tuset-Peiro, F. Vazquez-Gallego, J. Alonso-Zarate, L. Alonso, X. Vilajosana, Experimental energy consumption of frame slotted ALOHA and distributed queuing for data collection scenarios, Sensors 14 (8) (2014) 13416–13436. doi:10.3390/s140813416.
URL `http://www.mdpi.com/1424-8220/14/8/13416`
20. P. Tuset-Peiro, F. Vazquez-Gallego, J. Alonso-Zarate, L. Alonso, X. Vilajosana, LPDQ: A self-scheduled TDMA MAC protocol for one-hop dynamic low-power wireless networks, Pervasive and Mobile Computing 20 (2015) 84 – 99. doi:http://dx.doi.org/10.1016/j.pmcj.2014.09.004.
URL `http://www.sciencedirect.com/science/article/pii/S1574119214001576`
21. P. Tuset-Peiro, L. Alonso, F. Vazquez-Gallego, J. Alonso-Zarate, X. Vilajosana-Guillen, Demonstrating low-power distributed queuing for active {RFID} communications at 433 mhz, in: 2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2014, pp. 157–158. doi:10.1109/INFCOMW.2014.6849203.
22. M. Moradian, F. Ashtiani, Throughput analysis of a slotted aloha-based network with energy harvesting nodes, in: IEEE PIMRC, 2012, pp. 351–356. doi:10.1109/PIMRC.2012.6362809.
23. G. Yang, G.-Y. Lin, H.-Y. Wei, Markov chain performance model for IEEE 802.11 devices with energy harvesting source, in: IEEE GLOBECOM, 2012, pp. 5212–5217. doi:10.1109/GLOCOM.2012.6503948.
24. F. Iannello, O. Simeone, P. Popovski, U. Spagnolini, Energy group-based dynamic framed

ALOHA for wireless networks with energy harvesting, in: Annual Conference on Information Sciences and Systems (CISS), 2012, pp. 1–6. doi:10.1109/CISS.2012.6310797.

25. O. Briante, A. M. Mandalari, A. Molinaro, G. Ruggeri, F. Vazquez-Gallego, J. Alonso-Zarate, Duty-Cycle Optimization for Machine-to-Machine Area Networks based on Frame Slotted-ALOHA with Energy Harvesting Capabilities, in: European Wireless, 2014.
26. F. Iannello, O. Simeone, U. Spagnolini, Medium Access Control Protocols for Wireless Sensor Networks with Energy Harvesting, IEEE Transactions on Communications 60 (5) (2012) 1381–1389. doi:10.1109/TCOMM.2012.030712.110089.
27. F. Vazquez-Gallego, J. Alonso-Zarate, S. Wu, Y. Chen, K. Chai, Analysis and performance evaluation of dynamic frame slotted-ALOHA in wireless Machine-to-Machine networks with energy harvesting, in: IEEE GLOBECOM Workshop on Green Broadband Access, 2014.
28. F. Vazquez-Gallego, J. Alonso-Zarate, L. Alonso, Reservation Dynamic Frame Slotted-ALOHA for Wireless M2M Networks with Energy Harvesting, in: IEEE International Conference on Communications (ICC), 2015.
29. P. Lee, Z. A. Eu, M. Han, H. Tan, Empirical modeling of a solar-powered energy harvesting wireless sensor node for time-slotted operation, in: Wireless Communications and Networking Conference (WCNC), 2011 IEEE, 2011, pp. 179–184. doi:10.1109/WCNC.2011.5779157.
30. C. K. Ho, P. D. Khoa, P. C. Ming, Markovian models for harvested energy in wireless communications, in: Communication Systems (ICCS), 2010 IEEE International Conference on, 2010, pp. 311–315. doi:10.1109/ICCS.2010.5686445.
31. M. Gorlatova, A. Wallwater, G. Zussman, Networking low-power energy harvesting devices: Measurements and algorithms, in: INFOCOM, 2011 Proceedings IEEE, 2011, pp. 1602–1610. doi:10.1109/INFCOM.2011.5934952.
32. J. Norris, Markov Chains, Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press, 1998.
33. IEEE Std., IEEE Std 802.15.4-2006: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs) (September 2006).
34. CC2520 datasheet.
URL `http://www.ti.com/lit/ds/symlink/cc22520.pdf`