MAC Protocols for Wake-up Radio: Principles, Modeling and Performance Analysis

Debasish Ghose, *Student Member, IEEE,* Frank Y. Li, *Senior Member, IEEE,* and Vicent Pla

Abstract—In wake-up radio (WuR) enabled wireless sensor networks (WSNs), a node triggers a data communication at any time instant by sending a wake-up call (WuC) in an on-demand manner. Such wake-up operations eliminate idle listening and overhearing burden for energy consumption in duty-cycled WSNs. Although WuR exhibits its superiority for light traffic, it is inefficient to handle high traffic load in a network. This paper makes an effort towards improving the performance of WuR under diverse load conditions with a twofold contribution. We first propose three protocols that support variable traffic loads by enabling respectively clear channel assessment (CCA), backoff plus CCA, and adaptive WuC transmissions. These protocols provide various options for achieving reliable data transmission, low latency, and energy efficiency for ultra-low power consumption applications. Then, we develop an analytical framework based on an M/G/1/2 queue to evaluate the performance of these WuR protocols. Discrete-event simulations validate the accuracy of the analytical models.

Index Terms—IoT/WSNs, energy-efficient communication, WuR, MAC protocol, modeling and performance evaluation.

I. INTRODUCTION

The global Internet is shifting rapidly from connected computers to connected small devices, i.e., the Internet of Things (IoT). As one of the key enabling technologies, wireless sensor networks (WSNs) form an integral part of the IoT thanks to their low-power consumption and rapid deployment features. In many IoT applications, the energy-efficiency of IoT devices, e.g., battery-powered sensor nodes, is of paramount importance. Traditionally, duty-cycle medium access control (MAC) has been adopted in WSNs to reduce energy consumption by letting nodes sleep and wake up cyclically. However, duty-cycle MAC mechanisms suffer from idle listening and overhearing during their on-states. While idle listening occurs when a node listens to the channel for receiving control messages, overhearing occurs when a node overhears the control messages which are not intended to it. Although some of the proposed duty-cycle MAC protocols (e.g., PW-MAC [1] and RI-MAC [2]) are able to mitigate these problems from the transmitter end, idle listening cannot be

completely eliminated from the receiver end since a receiver has to listen to the medium for receiving beacon messages.

In recent years, a paradigm shift from duty-cycled WSNs to wake-up radio (WuR) has been envisaged. A WuR provides energy-efficient communication by diminishing overhearing and idle listening thanks to its superior energy performance. In a WuR enabled sensor node, an additional wake-up receiver (WuRx) is attached to the micro-controller unit (MCU) of the node to detect wake-up calls (WuCs). As demonstrated in [5] [6], the power consumption of a WuRx is 1000 times lower than that of the main radio, i.e., at the μW level versus mW for the main radio. Upon detection/reception of such a WuC sent by a transmitter, the WuRx triggers the MCU of the sensor node to wake up its main radio from the sleep mode. When the main radio is activated, data communication is performed according to the adopted MAC protocol. A WuR may function either in a transmitter-initiated or a receiver-initiated mode. Which mode suits better depends on the application requirements of a WuR based WSN. In general, receiver-initiated suits better for *data collection* whereas transmitter-initiated is more appropriate for event-triggered *data reporting*. The superiority of WuR over duty-cycle MACs was demonstrated in [5]. One of the primary advantages of WuR is that it works in an on-demand manner. Such an on-demand communication does not only increase energy efficiency but also reduces latency of data transmissions [5] [6]. In addition, the address decoding and matching abilities of WuRx exhibit its proficiency to process the received data in an energy-efficient way [7] [8]. That means, a WuRx wakes up its associated MCU only when the received WuC address has been validated, thus diminishing potential false wake-up and reducing energy consumption due to overhearing in the network.

In its initial form, WuR is designed for low traffic WSNs regardless of application scenarios. A typical application of WSNs is environment surveillance. Over time, applications have been expanded to more diverse areas, such as health monitoring, smart home and smart city, industrial automation, and potentially in WiFi. In certain scenarios, the offered traffic load in a WuR WSN could become very high due to suddenly increased packet generation rates. Despite enormous energy saving, timely reporting and low packet loss are essential if an abnormal event/behavior is observed. As a motivating example, consider a distributed WuR WSN deployed to monitor an industrial environment, such as a goods storage warehouse or a shipping harbor, and it is operated in the transmitter-initiated mode. If multiple nodes detect an abnormality, such as a fire, and report it at the same time, collisions may happen. Then heavy packet loss occurs since there is no mechanism adopted in WuR for collision avoidance among WuCs. On

D. Ghose and F. Y. Li are with the Department of Information and Communication Technology, University of Agder (UiA), 4898 Grimstad, Norway (email: debasish.ghose@uia.no; frank.li@uia.no).

V. Pla is with the ITACA, Universitat Politècnica de València (UPV), 46022 València, Spain (e-mail: vpla@upv.es).

TABLE I
A COMPARATIVE ANALYSIS OF OUR PROTOCOLS WITH THE STATE-OF-THE-ART WuR PROTOCOLS

Metric/Features	CCA-WuR [Ours]	CSMA-WuR [Ours]	ADPWuR [Ours]	DoRa/DC-DoRa [21]	OPWUM [20]	ALBA [6]	BoWuR [17]	Cor-WuR [5]
Network scenario	Single-hop	Single-hop	Single-hop	Single-hop	Multi-hop	Single-hop	Single-hop	Single- and multi-hop
Netw. sync. before WuC trans.	No	No	No	Yes	Yes	No	Yes	No
Multi-node competitions	Yes	Yes	Yes	No	No	No	Yes	No
Theoretical analysis	Yes	Yes	Yes	No	No	No	Yes	No
Simulation/testbed	Simulation	Simulation	Simulation	Simulation	Simulation	Testbed	No	Simulation
Single/dual radio	Single	Single	Single	Dual	Single	Single	Single	Single
Communication mode	TI	TI	TI	RI	TI and RI	—	TI	TI and RI
BO for WuC	No	Yes	Adaptive	No	Yes	No	Yes	No

the other hand, the traffic pattern of a WuR WSN may vary from time to time [10]. It is shown in [9] that WuR loses its superiority when traffic load is high. So far very few protocols which deal with WuC collisions can be found in the literature. Hence, it is of essential interest to reconsider WuR design and propose MAC protocols by taking traffic load into account. The aforementioned observations triggered our motivation to propose new WuR protocols in order to improve the performance of WuR under various traffic load conditions.

One essential reason for WuC collisions is that no clear channel assessment (CCA) before WuC transmission is performed for on-demand WuRs. In this paper, we propose three CCA enabled WuR protocols to improve the performance of WuRs. Furthermore, we develop a queuing model to evaluate the performance of WuR protocols and validate the accuracy of the model through extensive discrete-event simulations.

The main contributions of this work are as follows:

- Three MAC protocols are proposed for WuR-enabled WSNs: CCA-WuR, carrier sense multiple access (CSMA)-WuR and adaptive (ADP)-WuR.
- To model the behavior of the CCA-WuR, CSMA-WuR and ADP-WuR protocols, we develop a generic analytical framework based on an M/G/1/2 queue. The accuracy of the model is validated through discrete-event simulations.
- Closed-form expressions for calculating WuC loss probability, energy consumption, and latency are obtained based on the proposed analytical framework.
- To further improve the performance of WuR, two techniques are proposed. These techniques are targeted at increasing WuC transmission data rate and shortening WuRx addresses.

The remainder of this paper is structured as follows. In Section II, we summarize the related work and highlight the differences between our work and existing work. Section III presents the network scenario and assumptions. In Section IV, we present the three proposed protocols: CCA-WuR, CSMA-WuR and ADP-WuR. Section V presents the generic analytical framework for WuR protocols in details. Then the performance metrics are analyzed in Section VI. Section VII introduces the proposed techniques to improve the performance of WuR. Numerical results are presented in Section VIII, before the paper is concluded in Section IX.

II. RELATED WORK

The number of MAC protocols for WuR is growing in the literature. Two comprehensive surveys on the state-of-the-art WuR hardware, networking and MAC protocols were presented in [15] [16]. In brief, existing work falls into three main categories: 1) WuR circuit design; 2) WuR protocol design; and 3) performance evaluation of WuR. However, many studies cover two categories, e.g., circuit or protocol design plus performance evaluation.

A. WuR Circuit Design

The subcarrier modulation based correlator WuR [4] is one of the most popular inband WuR circuitry designs which uses a single channel to transmit both WuC and data. One of the key features of correlator WuR is that it can reach up to 100 meters which is the highest WuR transmission range among the designs that have been reported in the literature. ALBA-WUR [6] is another WuR which focuses on low-power consumption of WuRx. The authors presented the circuit design of ALBA-WUR and showed the effect of the data rate of WuC on the achieved transmission range. An ultra-low power, lower than 1 μW, WuR was reported in [7]. Its implementation can be adapted to different frequencies in the ISM band.

B. WuR Protocol Design

DoRa/DC-DoRa [21] is a WuR protocol which operates over two radios. For data transmission, the sink periodically sends a WuC addressed to each node, and the subsequent data transmission is performed on a separate channel. However, such a periodic polling mechanism increases transmission cost at the sink node. CMAC [22] is another multi-channel WuR protocol that uses a separate channel to send WuC and follows backoff (BO) before WuC transmission. In [20] the authors studied multi-hop WuR networks and proposed a protocol known as OPWUM. A node in OPWUM opportunistically selects the best relay node among its neighbors based on a given metric to resolve undesired neighborhood wake-up. For a similar reason, ZeroMAC [18] utilizes a radio frequency (RF) watchdog to wake up only the nodes on the communication path by sending unaddressed WuC in a hop-by-hop manner. RTM [19] and BoWuR [17] are two CSMA/collision avoidance (CA) alike WuR schemes which enable CCA plus BO before a WuC transmission.

C. Performance Evaluation of WuR

The superiority of Cor-WuR over duty-cycle MACs was demonstrated based on discrete-event simulations in [5]. However, the authors did not consider potential collisions for their performance evaluation. OPWUM [20] was validated

using computer simulations. Extensive simulations were performed to validate the DoRa/DC-DoRa protocols [21], showing their performance improvement over IEEE 802.15.4 and duty-cycle MAC protocols. Furthermore, ALBA-WUR was validated using a testbed. Its benefit over duty-cycle MAC was demonstrated based on a point-to-point topology using selective awakening. Both an analytical model and a testbed validation of WuR were presented in [8]. For their performance evaluation, the authors adopted a model based on an absorbing Markov chain. However, how to deal with WuC collisions is not reflected in that paper. The BoWuR [17] protocol was analyzed using a discrete time Markov chain and its performance improvement over WuR was shown under saturated traffic conditions.

In Table I, we summarize the major differences between our protocols and a few other state-of-the-art WuR protocols. None of the WuR protocols that have been reported in the literature considered purely asynchronous networks for tackling WuC collisions. Rather, the existing WuR protocols either perform network synchronization before WuC transmissions, adopt a synchronous mode, or employ CSMA/CA (IEEE 802.11) MAC to handle WuC collisions. For instance, multiple nodes in a BoWuR based network count down their BO counters from the same time instant, the same as what is used in WiFi where multiple stations start counting down after a distributed coordination function (DCF) interframe space. Recall that WuR works in an on-demand and purely asynchronous manner and the MCU of a WuR-enabled node only wakes up when it generates a packet. The clusterhead does not adopt any mechanism to keep track of such an on-demand wake-up pattern. Therefore, it is infeasible to exchange periodic/aperiodic beacons for maintaining network synchronization, especially in transmitter-initiated mode which is the focus of this work. In this paper, we propose three MAC protocols which are tailored to WuR-enabled WSNs/IoT networks operated in an asynchronous mode and develop an analytical framework to evaluate the performance of these protocols.

III. NETWORK SCENARIO AND WUR DESCRIPTION

In this section, we first describe the network scenario and assumptions and then present the design of a reference WuR.

A. Network Scenario and Assumptions

Star and tree topologies are two popular network topologies that are used for environmental surveillance in typical battery-powered WSN applications. In such a network, sensor nodes monitor the environment in a specific sensing area and transmit their measured/monitored data towards one common destination, the sink or clusterhead. In this study, we consider an *event-triggered data reporting* WSN with a star topology since such a scenario is more prone to transmission collisions. The sensor nodes are WuR-enabled and are operated in the transmitter-initiated mode. Collisions occur if the transmissions of more than one node overlap with each other.

Under such a scenario, consider a network cluster consisting of $N + 1$ sensor nodes including one clusterhead and N member nodes, as shown in Fig. 1. The N member nodes
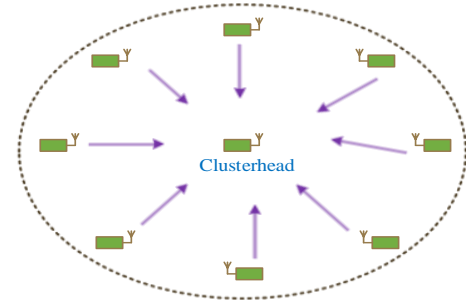

Fig. 1. A WSN with multiple contending nodes and a sink.

compete with each other *in an asynchronous mode* for data reporting towards the clusterhead over a single hop. Each node has a finite queue capacity and is equipped with a WuR transceiver in addition to its main radio.

Assume that at each node (except the clusterhead) packets are generated based on a Poisson process with an arrival rate of λ. The channel is considered to be error-free and no hidden terminal exists in this cluster.

B. Description of the Reference WuR Prototype

The reference WuR design considered in this study is based on the subcarrier modulation WuR introduced in [3]. Its design details can be found in [3] [4]. A CC1101 [25] chip, which operates in the 868 MHz ISM band and supports different modulation techniques including on-off keying, is used as the main radio of a node. To extend its transmission range, an *additional* CC1190 RF front-end [4] is integrated at the transmitter, reaching a transmission power level of +20 dBm. A 125 kHz wake-up signal is modulated using on-off keying modulation on the 868 MHz carrier frequency at the transmitter. At the transmitter side, there is only one channel through which both data and WuC are transmitted from the main radio by switching the antenna alternatively.

At the receiver side, a WuRx is built based on an off-the-shelf low-frequency integrated circuit, AS3932 [27], that works at 125 kHz with address decoding and matching capabilities. An on-off keying modulated WuC is demodulated at the receiver using a Schottky diode, followed by a low-pass filter. Afterward only the envelope signal is processed at the WuRx. Upon detecting a wake-up signal, it interrupts the MCU to switch from the sleep to the active mode. The WuC frame structure is based on AS3932 [27], consisting of a carrier burst, a preamble (0101010.... ON/OFF modulated carrier), and a 16-bit address.

IV. TRANSMISSION PROTOCOLS FOR WUR

In this section, we present the design and working principle of a reference WuR and then propose three MAC protocols with a focus on collision avoidance among WuCs.

A. Correlator-WuR (Cor-WuR): A Benchmark Protocol

The reference WuR protocol considered in this paper is Cor-WuR, also refereed to as SCM-WuR in [5], [11] and [17]. It can operate either in the transmitter-initiated or the receiver-initiated mode. In what follows, we focus on a Cor-WuR
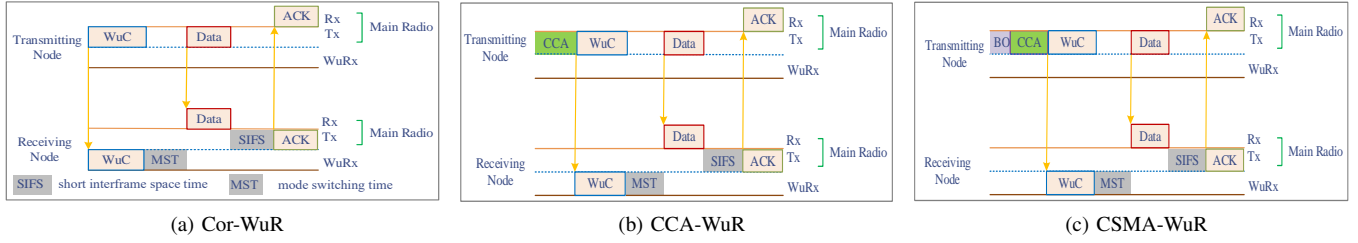
Fig. 2. Illustration of the Cor-WuR, CCA-WuR, and CSMA-WuR protocols.

based WSN operated in the transmitter-initiated mode. A data transmission cycle of the transmitter-initiated WuR lasts from the instant of a WuC initiation to the instant when an acknowledgment (ACK) is received, as illustrated in Fig. 2(a). In the transmitter-initiated mode, a node which has a packet to transmit sends a WuC to the targeted node using its main radio. After receiving the WuC and decoding the address correctly, the targeted node switches on its main radio from the sleep mode for data communication. Right after a data exchange finishes, both transmitter and receiver switch off their main radios and go to sleep. However, both nodes keep their WuRxs active and continuously listen to the channel. It is worth mentioning that in Cor-WuR no ACK is sent upon the successful reception of the WuC [5] [11]. Neither does it exist any MAC mechanism for WuC transmissions. For the subsequent data transmission after receiving a WuC correctly, a MAC protocol, for instance CSMA/CA, could be adopted. Depending on the MAC protocol being adopted, the main radio of the sending node may or may not perform CCA or/and a BO procedure before a packet transmission.

In the rest of the section, we propose three MAC protocols for WuR to avoid WuC collisions and explain their principles.

B. CCA Enabled WuR (CCA-WuR)

As mentioned earlier, the existing WuR solutions do not perform carrier sensing prior to a WuC transmission. The main idea of a CCA enabled WuR is to make sure that no other node is transmitting a WuC before it starts sending its own WuC. Such a CCA mechanism reduces collision among WuCs. When an event is detected by a WuR integrated sensor node, it performs CCA first to check whether the channel is idle or not. If the WuR finds the channel idle for a CCA duration, it transmits the WuC; otherwise it performs CCA again. This procedure will continue until the maximum number of attempts is reached. If the maximum number of attempts is reached, the data frame will be discarded from the queue.

Since the WuC is sent through a wireless medium, all neighboring WuRxs receive the same WuC. After decoding the WuC frame, the targeted WuR, i.e., the clusterhead switches on its main radio for communication and the main radio of the other nodes will continue to sleep. When both transmitting and receiving main radios are switched on, data exchange starts immediately. The transmission cycle ends up with an ACK from the receiving main radio to the transmitting main radio. At the end of each transmission cycle, the main radios of both nodes go to sleep but their WuRxs are still actively listening to the channel. The principle of CCA-WuR is shown in Fig. 2(b) for one data communication cycle.

C. CSMA-CA Enabled WuR (CSMA-WuR)

Although CCA-WuR is capable of handling light traffic load in WuR-based WSNs, performing CCA alone is not sufficient to eliminate collisions among WuCs during a high traffic load episode. To overcome the limitation of CCA-WuR, we further propose CSMA-WuR, a CSMA-CA enabled WuR MAC protocol. CSMA-WuR works similarly to the unslotted CSMA-CA MAC protocol of IEEE 802.15.4 but it is tailored to WuCs. It is worth mentioning that the main difference between the CSMA/CA and the CSMA-CA protocol is the order of performing CCA and BO. In CSMA/CA, *a node performs CCA first and then BO afterward, whereas the order is reversed in CSMA-CA*.

In CSMA-WuR, upon detecting an event by a node, it first performs a BO procedure without checking whether the channel is idle or not. As soon as the BO waiting time ends, it checks the channel status by performing a CCA. If it finds the channel idle for a duration of CCA, it sends a WuC. Otherwise it repeats the BO and CCA procedure again. Similar to CCA-WuR, this procedure will be repeated until the attempt limit has been reached. The rest of the CSMA-WuR operation is the same as in CCA-WuR. The whole operation procedure of the CSMA-WuR protocol is presented in Fig. 2(c).
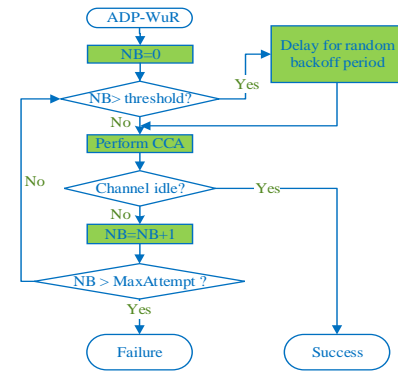

Fig. 3. The working procedure of the ADP-WuR protocol.

D. Adaptive WuR (ADP-WuR)

From the design principles of CCA-WuR and CSMA-WuR, it is clear that CCA-WuR and CSMA-WuR are well suited for light and heavy traffic respectively. However, traffic load varies over time and it might not be beneficial to always employ either CCA-WuR or CSMA-WuR *alone* under all traffic conditions. Therefore, we further propose an adaptive protocol which enables either CCA-WuR or CSMA-WuR adaptively based on traffic conditions. To do so, each node needs to keep track on its WuC transmission attempt counter

and compares it with a pre-configured attempt threshold for MAC mechanism adaptation.

Fig. 3 illustrates the working procedure of ADP-WuR, where NB stands for the BO stage. Considering a static network cluster, we set a pre-configured attempt threshold in this study. Initially, a node follows the CCA-WuR mechanism for a WuC transmission. If the transmission is unsuccessful up to the pre-configured threshold (i.e., NB), it switches to CSMA-WuR for its next attempts of the ongoing WuC transmission. Let us consider the attempt threshold as two. Whenever a node detects an event, it follows CCA-WuR first. After two unsuccessful attempts, the node adopts CSMA-WuR from its third attempt onward until the limit is reached.

Furthermore, the attempt threshold of a node may also be configured dynamically based on traffic conditions or/and transmission status. To do so, each node has to keep the statistics of its previous transmissions (e.g., WuC loss, successful packet delay, number of attempts required for previous successful WuC transmissions). Such a dynamic threshold provides more flexibility for MAC operations at the cost of high complexity. On the other hand, for event-triggered data reporting, congestion may arise abruptly and releases after a short period. In this case, the threshold adaptation scheme may not be able to keep up with the instantaneous traffic conditions and thus may not achieve the expected performance improvement.
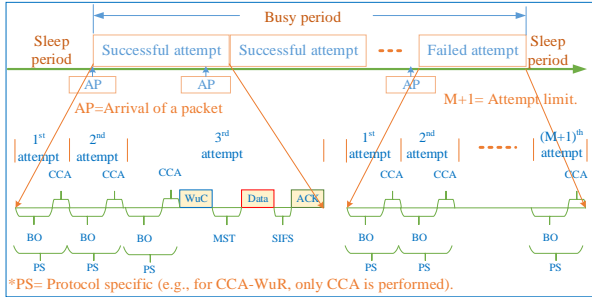

Fig. 4. Illustration of a busy period based on the proposed WuR protocols.

Fig. 4 illustrates a generic busy period of CCA-WuR, CSMA-WuR and ADP-WuR. A busy period is defined as the period that begins with the arrival of a packet into an empty queue and ends when the queue becomes empty again. Note that all packets generated during a busy period will be processed (either successfully transmitted or discarded) in that busy period. In other words, a busy period may consist of multiple packets, with one or more transmission attempts per packet as shown in this figure. A successful attempt means that a packet is successfully delivered with an ACK received at the transmitter, whereas a failed attempt means that a packet is discarded after the retry limit has been reached. The lower half of this figure illustrates the behavior of the transmitter for a successful and a failed attempt (note that the detailed behavior within a transmission attempt is protocol dependent. As an example, CSMA-WuR which requires BO plus CCA is shown in Fig. 4 for illustration purposes).

V. MODELING WuR PROTOCOLS

The analytical framework for modeling WuR and the proposed protocols is presented in this section. This framework was initially inspired by the approach presented in [12], which was based on the analysis of a regenerative cycle of the M/G/1 queue. In [12], the authors considered BO first and CCA afterward and accordingly calculated packet loss probability based on a similar busy period definition presented therein. However, their model was based on an *infinite size queue* and assumed the same packet processing rate regardless of the BO period structure of the adopted MAC protocol. A consequence of their assumption is that the key performance metrics have the same value irrespectively of the BO periods of the protocol under study. Furthermore, Cor-WuR, which is the benchmark for our protocol design, does not perform BO or CCA prior to WuC transmissions. Hence, the presented model in [12] is not directly applicable for performance evaluation of Cor-WuR.

Considering a *finite size queue*, we develop below a generic framework for performance evaluation of both Cor-WuR and the proposed protocols based on an M/G/1/2 queuing model which captures the on-demand nature of the WuR operation. Unlike the M/G/1 model, our model considers that there can be at most two packets in the queue, i.e., the one at the head of the queue (HoL), whose transmission is underway, plus another one. This choice is motivated by the fact that sensors usually have small buffers, and also because a reduced buffer size model (e.g., buffer-less; or equivalent formulations) is a technique that has been adopted to extend Bianchi's model [13] to a non-saturated scenario (see, e.g., [14]). The validity and accuracy of this modeling approach will be verified through extensive computer simulations (see Sec. VIII). The notations of the analytical model are summarized in Table II.

A. Models for CCA-WuR, CSMA-WuR and ADP-WuR

In our model, we focus on a single representative device, referred to as the *tagged device*. The tagged device is modeled as an M/G/1/2 queue in which the service time of a packet represents the duration from the epoch that the packet arrived at the HoL to the epoch it is transmitted successfully with an ACK or it is discarded.

The analysis of the model revolves around the calculation of the probability of an unsuccessful attempt, i.e., the probability that the channel is sensed as busy right after a CCA. Denote this probability as α and assume that it is constant regardless of the attempt stage. A similar assumption is also adopted in many MAC protocol studies, including [13]. Based on this assumption, it follows that the service time is independent and identically distributed.

Using the same reasoning as in [12] we can write

$$\alpha = \frac{(N-1)(1-P_L)\,\mathrm{E}[\Gamma](T_{\mathrm{CCA}} + T_{\mathrm{TA}})}{1/\lambda + \mathrm{E}[\Gamma]\,\mathrm{E}[D_{\mathrm{HoL}}]}, \qquad (1)$$

where Γ is the number of packets served (note that served includes both successfully transmitted and acknowledged, and discarded after the maximum number of attempts, $M+1$) in a busy period; P_L is the probability of a packet being discarded after $M+1$ unsuccessful attempts; $T_{\mathrm{TA}} = T_{\mathrm{wuc}} + T_{\mathrm{MST}} + T_{\mathrm{data}} +$

TABLE II
NOTATIONS USED IN THE ANALYTICAL FRAMEWORK

Notation	Description
N	Number of nodes in the network/network cluster
Q	Queue capacity of a node (in packets)
$M+1$	Maximum number of attempts incl. 1 initial transmission and M attempts
W_i	Contention window size in i^{th} contention stage
λ	Packet generation rate to a node
α	Probability that the channel is sensed as busy right after each CCA
$E[\Gamma]$	Expected number of packets served in a busy period
P_S	Successful packet transmission probability
P_L	WuC loss probability
a_0	Probability that no packet arrives during the time a packet is at the HoL
w_k	Mean accumulated duration including BO and CCA for the k-th attempt
$E[D_{HoL}]$	Mean sojourn time of a packet at the HoL
T_{FA}	Duration of a failed transmission attempt
T_{SA}	Duration of a successful transmission attempt
$E[T_{BP}]$	Mean duration of a busy period
$E[A]$	Expected number of transmission attempts per frame

$T_{SIFS} + T_{ack}$ is the total duration of a transmission attempt, including the reception of ACK; and D_{HoL} is the duration from the time a packet arrives at the HoL until right after the last CCA for that packet (after the last CCA the packet will be either transmitted or discarded).

In what follows, we derive expressions for P_L, $E[D_{HoL}]$, and $E[\Gamma]$, which in turn depend on α. Thus, when substituted into (1) a non-linear equation will be obtained from which the value of α can be solved numerically. Clearly,

$$P_L = \alpha^{M+1}. \tag{2}$$

Let W_i and σ denote, respectively, the contention window size at the i-th BO stage and the duration of a BO slot. The mean accumulated duration of the BO plus CCA until the k-th transmission attempt, w_k, is given by

$$w_k = \sum_{i=0}^{k-1} \frac{W_i - 1}{2}\sigma + kT_{CCA}. \tag{3}$$

Then, the expected HoL delay can be obtained as

$$E[D_{HoL}] = \sum_{v=0}^{M} \alpha^v (1-\alpha) w_{v+1} + \alpha^{M+1} w_{M+1}. \tag{4}$$

The first part of (4) corresponds to a situation where the transmission, including WuC and data, is successful, whereas the second part corresponds to a situation where a WuC is discarded after $M+1$ unsuccessful CCA attempts.

At the time instant a packet reaches the HoL, it will be the only packet in the queue. Thus, if no packet arrives during its sojourn at the HoL, that packet will be the last one in a busy period (see Fig. 4). Let us denote by a_0 the probability of this event, i.e, that no packet arrives during the time a packet is at the HoL. Then, the number of packets served in a busy period, Γ, follows a geometric distribution

$$P(\Gamma = k) = (1-a_0)^{k-1}a_0, \qquad k = 1, 2, \ldots, \tag{5}$$

and

$$E[\Gamma] = \frac{1}{a_0}. \tag{6}$$

Let us now represent by S the sojourn time of a packet at the HoL, i.e., the time from its arrival at the HoL until it is successfully transmitted (including the reception of ACK) or discarded. Note that $S = D_{HoL}$ if the packet is discarded, and

$S = D_{HoL} + T_{TA}$ if it is transmitted. Furthermore, let $f_S(t)$ denote the probability density function of S. Then, recalling that the packet generation process is Poisson, we can write

$$a_0 = \int_0^\infty f_S(t)e^{-\lambda t}\, dt = \mathcal{L}\{f_S\}(\lambda), \tag{7}$$

where $\mathcal{L}\{f_S\}(\lambda)$ denotes the Laplace transform of f_S evaluated at λ.

Using Laplace transforms it can be shown that (the details of this deduction are omitted due to the page limit)

$$a_0 = \sum_{v=0}^{M} \alpha^v (1-\alpha) H_{v+1}(\lambda) e^{-T_{TA}\lambda} + \alpha^{M+1} H_{M+1}(\lambda), \tag{8}$$

where

$$H_n(s) = \frac{e^{-nT_{CCA}s}}{(1-e^{-\sigma s})^n} \prod_{i=0}^{n-1} \frac{1-e^{-W_i\sigma s}}{W_i}. \tag{9}$$

Although the numerical evaluation of (8) does not pose a problem, we propose herein an alternative approach to obtain a_0, based on an approximation, that leads to a simpler derivation and expression. The approximation consists of substituting the random duration of each BO stage by its mean value. Using this approximation, it easily follows that

$$a_0 \approx \sum_{v=0}^{M} \alpha^v (1-\alpha) e^{-(w_{v+1}+T_{TA})\lambda} + \alpha^{M+1} e^{-w_{M+1}\lambda}. \tag{10}$$

As noted, substituting (2), (4), (6) (along with (8) or (10)) into (1) yields a non-linear equation from which the value of α can be solved numerically.

Next we particularize the coefficients w_k (defined in (3)) for each of the considered protocols and introduce a minor adaptation of the analysis for the case of WuR.

1) CCA-WuR: CCA-WuR performs CCA prior to a WuC transmission but not BO. This is equivalent to considering a contention window of size equal to 1 regardless of the attempt stage, i.e., $W_i = 2^0 = 1$ for $i = 0, 1, \ldots, M$. Then,

$$w_k = kT_{CCA}. \tag{11}$$

2) CSMA-WuR: Prior to WuC transmission, the CSMA-WuR protocol performs BO and then CCA. In all attempt stages the random number of slots is selected uniformly from $\{0, 1, \ldots, W-1\}$, i.e., the size of the contention window is kept constant: $W_i = W$ for $i = 0, 1, \ldots, M$.

Hence, w_k can be calculated as

$$w_k = k\left(\frac{W-1}{2}\sigma + T_{CCA}\right). \tag{12}$$

3) ADP-WuR: The ADP-WuR protocol is a combination of the two previous ones. In the first t attempts it behaves as CCA-WuR, and as CSMA-WuR after that. That is,

$$W_i = \begin{cases} 1 & \text{if } i = 0, \ldots, t-1 \\ W & \text{if } i = t, \ldots, M+1, \end{cases} \tag{13}$$

and then

$$w_k = \begin{cases} kT_{CCA} & \text{if } k = 0, \ldots, t-1 \\ (k-t+1)\frac{W-1}{2}\sigma + kT_{CCA} & \text{if } i = t, \ldots, M+1. \end{cases} \tag{14}$$

B. Model for Cor-WuR

According to Cor-WuR, a node sends a WuC to the intended WuRx whenever it generates a data packet. Unlike in the above analyzed protocols, collisions with the transmission of other devices are more likely to occur since no CCA is performed. Thus, it is essential to investigate the collision probability.

Let us denote by α the collision probability. As before, α represents the probability of an unsuccessful transmission attempt, but now the transmission (of the WuC and the data frame that follows) actually occurs and the failure is detected when the ACK timer times out.

After a failed attempt, a device starts over the transmission process until the maximum number of attempts, $M + 1$, has been reached. If the transmission could not be completed successfully after the maximum number of attempts, the device simply discards that data packet. The illustration of a frame transmission is presented in Fig. 5, where $T_{\text{FA}} = T_{\text{wuc}} + T_{\text{MST}} + T_{\text{data}} + T_{\text{SIFS}}$ (respectively, $T_{\text{SA}} = T_{\text{TA}}$) is the duration of a failed (respectively, successful) transmission attempt.

A transmission attempt of the tagged device will be successful if no busy periods of the other $N - 1$ devices overlap with its. For this to occur, none of the other $N - 1$ devices should start a busy period during the vulnerable period of the transmission of the tagged device. For simplicity, we assume that all busy periods have a constant duration equal to its mean, $\text{E}[T_{\text{BP}}]$. With this assumption, an approximation for the collision probability can be obtained as follows

$$1 - \alpha \approx \left(e^{-\frac{\lambda}{\text{E}[\Gamma]}(\text{E}[\Gamma]\,\text{E}[A]T_{\text{FA}} + T_{\text{SA}})} \right)^{N-1}. \quad (15)$$

In this approximation, it is also assumed that busy periods occur according to a Poisson process with rate $\lambda / \text{E}[\Gamma]$.

The mean duration of a busy period is given as

$$\text{E}[T_{\text{BP}}] = \text{E}[\Gamma]\,\text{E}[A]T_{\text{FA}}, \quad (16)$$

where A is the number of transmission attempts per frame, and its mean value is obtained as follows

$$\text{E}[A] = \sum_{i=0}^{M} \alpha^i (1 - \alpha)(i + 1) + \alpha^{M+1}(M + 1) = \frac{1 - \alpha^{M+1}}{1 - \alpha}. \quad (17)$$

Since the WuC is sent without a prior BO or CCA, the probability that no packet arrives during a packet sojourn at the HoL can be approximated as

$$a_0 \approx \sum_{v=0}^{M} \alpha^v (1 - \alpha)e^{-(vT_{\text{FA}} + T_{\text{SA}})\lambda} + \alpha^{M+1}e^{-(M+1)T_{\text{FA}}\lambda}. \quad (18)$$

As no retransmission (after a failed attempt) is considered in Cor-WuR, (18) is simplified by setting $M = 0$

$$a_0 \approx (1 - \alpha)e^{-T_{\text{SA}}\lambda} + \alpha e^{-T_{\text{FA}}\lambda}, \quad (19)$$

which can be further simplified if we assume that $T_{\text{TA}} = T_{\text{SA}} \approx T_{\text{FA}}$:

$$a_0 \approx e^{-T_{\text{TA}}\lambda}. \quad (20)$$

Similarly, $\text{E}[A] = 1$. Substituting these simplified equations into (15) yields

$$\alpha \approx 1 - e^{-(N-1)\lambda T_{\text{SA}}(1 + e^{-T_{\text{TA}}\lambda})}. \quad (21)$$
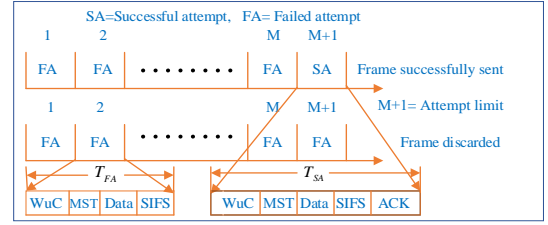

Fig. 5. A transmission cycle of Cor-WuR.

VI. PERFORMANCE ANALYSIS

To evaluate the performance of the WuR MAC protocols, we define three performance metrics and derive their expressions based on the obtained unsuccessful attempt probability.

A. WuC Loss Probability

The WuC loss probability, denoted by P_L, is defined as the probability of a packet being discarded after $M + 1$ unsuccessful attempts. The WuC loss probability has already been introduced in Sec. V.A and its general expression is given in (2). Note, however, that the mean accumulated duration including BO and CCA for the k-th attempt, w_k, which is needed for P_L calculation, is protocol specific.

Since Cor-WuR discards a packet after one unsuccessful attempt, P_L can be obtained directly using (21), i.e., $P_L = \alpha$.

B. Average Packet Delay

The average delay, denoted by T_S, is defined as the duration from the time a packet arrives at the HoL until it is successfully transmitted or discarded. This means that the delay experienced by an unsuccessful packet before it is discarded has been counted in the expressions derived below. Accordingly, T_S can be calculated as

$$T_S = (1 - P_L)T_t + P_L T_L, \quad (22)$$

where T_t and T_L are the mean delays from the time a packet arrives at the HoL until it is successfully transmitted (including the reception of ACK) and lost respectively. T_L is given as

$$T_L = \sum_{i=0}^{M} \frac{W_i - 1}{2}\sigma + (M + 1)T_{\text{CCA}}.$$

The mean HoL delay can be written as $\text{E}[D_{\text{HoL}}] = P_L T_L + (1 - P_L)(T_t - T_{\text{TA}})$, where the $(T_t - T_{\text{TA}})$ term corresponds to the HoL delay for a successfully transmitted packet. Now, T_t can be obtained as

$$T_t = \frac{\text{E}[D_{\text{HoL}}] - P_L T_L}{1 - P_L} + T_{\text{TA}}.$$

For a given protocol, $\text{E}[D_{\text{HoL}}]$ and T_L can be formulated based on the protocol behavior (e.g., for CCA-WuR, we have $T_L = (M + 1)T_{\text{CCA}}$).

The Cor-WuR protocol does not perform CCA or BO. Correspondingly, we have $T_S = P_L T_{\text{FA}} + (1 - P_L)T_{\text{SA}}$.

C. Energy Consumption

The energy consumption is quantified by E_S (in Joule), which represents the average energy consumed by the tagged device when it attempts to transmit a packet. This attempt can result in a successful transmission or a failed one. Let E_{wuc}, E_{data}, E_{ack}, E_{CCA}, E_{SIFS}, E_{idle}, E_{bo}, and E_{MST} be respectively the energy consumptions for WuC transmission, data packet transmission, ACK reception, CCA, SIFS, idle slot, BO, and switching on the MCU. Then E_S is given as

$$E_S = (1 - P_L)E_t + P_L E_L, \tag{23}$$

where E_t and E_L are the energy consumed for a successful and lost packet respectively, and E_L can be calculated as

$$E_L = \sum_{i=0}^{M} \frac{W_i - 1}{2} E_{\text{bo}} + (M + 1)E_{\text{CCA}}.$$

The average energy consumed by a packet during BO and CCA, denoted as E_{HoL}, can be formulated as

$$E_{\text{HoL}} = P_L E_L + (1 - P_L)(E_t - E_{\text{TA}}), \tag{24}$$

where $E_{\text{TA}} = E_{\text{wuc}} + E_{\text{MST}} + E_{\text{data}} + E_{\text{SIFS}} + E_{\text{ack}}$ and $(E_t - E_{\text{TA}})$ is the energy consumed by a successful packet during BO and CCA. Now, E_t can be calculated as

$$E_t = \frac{E_{\text{HoL}} - P_L E_L}{1 - P_L} + E_{\text{TA}}, \tag{25}$$

where

$$E_{\text{HoL}} = \sum_{v=0}^{M} \alpha^v (1 - \alpha) \sum_{i=0}^{M} \frac{W_i - 1}{2} E_{\text{bo}} + (M + 1)E_{\text{CCA}} + \alpha^{M+1} E_L.$$

Note that E_L and E_{HoL} depend on the behavior of each specific protocol (e.g., for CCA-WuR, $E_L = (M + 1)E_{\text{CCA}}$ and $E_{\text{HoL}} = (M + 1)E_{\text{CCA}} + \alpha^{M+1} E_L$).

As mentioned earlier, Cor-WuR does not perform CCA or BO. Therefore, $E_S = P_L E_{\text{FA}} + (1 - P_L)E_{\text{SA}}$ where $E_{\text{SA}} = E_{\text{TA}}$ and $E_{\text{FA}} = E_{\text{wuc}} + E_{\text{MST}} + E_{\text{data}} + E_{\text{SIFS}}$.

VII. EFFECT OF DATA RATE AND ADDRESS SCHEME

In this section, we present two techniques for performance improvement of WuR protocols obtained from our MAC protocol design experience.

A WuC consists of a carrier burst, preamble, and 16-bit address scheme. Since a WuC is modulated using on-off keying and transmitted at a lower data rate, the duration of a WuC is indeed longer than that of a data packet. For example, the time needed to transmit a data packet with a size of 35 bytes at a data rate of 250 kbps is 1.12 ms [12], whereas the WuC transmission time is 12.2 ms [5]. Due to such a long WuC transmission, the busy period increases, resulting in a reduced successful packet transmission probability. One of the effective ways to shorten WuC duration is to increase the date rate for WuC transmissions. It is worth mentioning that the benchmark Cor-WuR [5] adopted 2730 bps for its WuC transmission. We recommend therefore to increase the data rate of WuC transmissions, to, e.g., 5460 bps. However, a tradeoff between WuC duration and coverage needs to be considered before deciding which data rate to adopt.

TABLE III
PARAMETER CONFIGURATION [5] [11] [17] [26] [27]

Radio type	Parameter	Value	Unit
Common	Supply voltage	3	V
Main radio	Data rate	250	kbps
	Transmission current	17.4	mA
	Reception current	18.8	mA
	Idle current	20	μA
	SIFS duration	192	μs
	Payload size	35	bytes
	ACK frame size	11	bytes
Wake-up radio	WuC duration	6, 12.2	ms
	WuC Transmission current	152	mA
	Reception current (WuRx)	8	μA
	Sleep current	3.5	μA
	BO current	5.16	mA
	CCA current	20.28	mA
	MCU switching current	2.7	μA
	Time to switch on MCU	1.79	ms
	CCA duration	1.92	ms
	Slot time	320	μs
	Contention window size	32, 64	slots
	WuC packet size	4, 2	bytes
	Maximum WuC attempts	7	times

Another point for performance improvement is to apply a shorter address scheme. The IEEE 802.15.4 standard adopts a 16-bit address scheme to cover up to 2^{16} nodes. However, such a network size is not common for small or medium size WuR enabled WSNs since it increases WuC delay and energy cost significantly. With a shorter address, the busy period shrinks and the successful WuC transmission probability increases. For a network cluster composed of up to 256 nodes, we would configure the address length to 8 bits.

VIII. NUMERICAL RESULTS AND DISCUSSIONS

In this section, we provide numerical results to evaluate the performance of the WuR protocols. The accuracy of the analytical model has been verified by discrete-event simulations. In all cases, the discrepancy between the analytical results and those obtained by simulations was below 2%. Note that *ana* and *sim* in these figures indicate the results obtained from analytical expressions and simulations respectively.

Consider a WuR WSN as shown in Fig. 1, with an average packet arrival rate of $\lambda = 10$ packets/s and variable traffic load represented by different number of nodes, $N \in \{10, 15, ..., 30\}$, in the network. The remaining parameters are configured based on the specifications listed in Table III.

To perform computer simulations, we made a custom-built discrete-event simulator. The developed simulator mimics the behavior of the studied WuR MAC protocols according to the principle of each protocol presented above. That is, a WuR-enabled node wakes up upon the arrival of a packet into the queue and maintains a timestamp to keep track the time duration of every single state transition. The sending node follows the procedure of the adopted MAC protocol, i.e., Cor-WuR, CCA-WuR, CSMA-WuR, or ADP-WuR prior to each WuC transmission. Upon receiving a WuC, the clusterhead, which is the only targeted receiver in our scenario, decodes and validates the address of the WuC. If the decoded address matches its own, it switches on its main radio for data communication. The timestamp for each packet is maintained
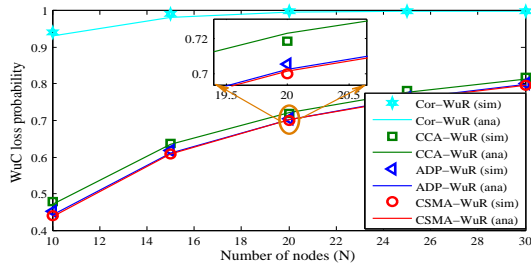
Fig. 6. WuC loss probability comparison of Cor-WuR, CCA-WuR, CSMA-WuR, and ADP-WuR with the WuC duration as 12.2 ms.
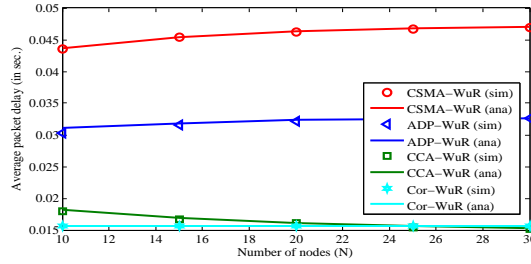

Fig. 8. Comparison of proposed protocols in terms of average delay for a successful packet when the WuC duration is 12.2 ms.

Fig. 7. Average packet (successful/discarded) delay comparison of Cor-WuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

Fig. 9. Energy consumption comparison of Cor-WuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

until the packet has been successfully delivered or is discarded. Note that the simulation results are completely independent of the analytical model and expressions presented earlier in Section V.

A. WuC Loss Probability: A Comparison of Four Protocols

Fig. 6 presents the P_L variation as the network size, N, varies. From this figure, it is evident that the analytical results match precisely the simulation results. With a large margin, all three proposed protocols outperform the Cor-WuR protocol in terms of WuC loss probability, P_L. The reason is that Cor-WuR does not have a defer mechanism for WuC transmissions. On the contrary, our proposed protocols enable CCA, or BO plus CCA, or adaptive mechanism prior to WuC transmissions, thus eliminating collisions. Therefore, data transmissions with higher reliability can be achieved based on the proposed protocols.

On the other hand, P_L increases with the number of nodes. When traffic load increases in the network, the duration of busy periods increases, leading to a lower successful WuC transmission probability. A similar trend applies to all the studied protocols as the network size grows. Among CCA-WuR, CSMA-WuR, and ADP-WuR, the latter two perform slightly better than CCA-WuR, thanks to the additional BO procedure. The CSMA-WuR and ADP-WuR show nearly similar performance all the studied network size. This is due to the fact that the ADP-WuR performs BO in addition to CCA when the transmission when pre-configured threshold exceeds.

B. Average Packet Delay

Fig. 7 illustrates how the average delay for a successful or discarded packet varies with different network sizes. From the
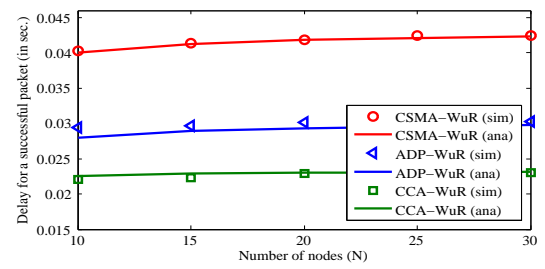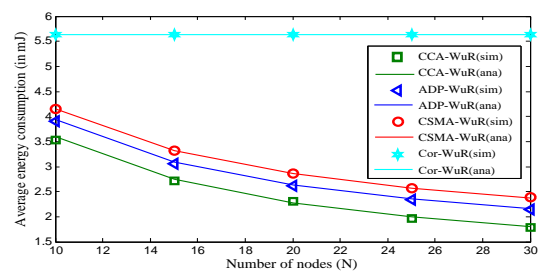
figure it is clear that T_S of Cor-WuR is the shortest one among all four protocols and it is constant regardless of the network size. This behavior is mainly due to the fact that Cor-WuR neither performs CCA or BO prior to a WuC transmission nor does it allow any retransmissions. On the contrary, our proposed protocols perform CCA or CCA plus BO before WuC transmissions for both initial and retry attempts. Such CCA or BO plus CCA and retry stages introduce extra delay for WuC transmissions. Among the proposed three protocols, the average packet delay of CCA-WuR is the shortest since it needs only CCA before a WuC transmission. Between CSMA-WuR and ADP-WuR, the latter one performs better since it performs BO (prior to CCA) only after the pre-configured attempt threshold is reached. On the other hand, the average packet delay, T_s, of CSMA-WuR and ADP-WuR becomes longer with a larger network size, whereas an opposite trend is observed for the delay of CCA-WuR. The reason is as follows. The CCA-WuR protocol keeps a packet in the queue for only up to 7 CCA intervals, (i.e., $7 \times T_{CCA} = 13.44$ ms) before discarding the packet and this duration is shorter than the duration of a successful packet transmission. More specifically, a successful transmission (even if it occurs after the first CCA, i.e., $T_{CCA} + T_{TA} = 17.382$ ms where $T_{CCA} = 1.92$ ms and $T_{TA} = 12.2$ ms + 1.79 ms + (35+11)×8 bits/250 kbps = 15.462 ms) takes more time than discarding the packet after seven unsuccessful CCA attempts. As observed in Fig. 6, the proportion of packets that are discarded, P_L, increases with network size. As a consequence, the average packet delay decreases.

Fig. 8 represents the average delay for transmitting a packet successfully based on the proposed protocols. It is observed that CCA-WuR performs better than the other two protocols since no BO is performed in CCA-WuR. When comparing the

delay and loss probability performance of Cor-WuR versus the proposed protocols as shown in Fig. 6 - Fig. 8, we need to keep in mind that Cor-WuR, which achieves the shortest delay, does not allow retransmissions. This means that each generated packet will be transmitted only once, regardless of its transmission history. Therefore, employing Cor-WuR directly could lead to a disadvantage that no packets will be transmitted successfully when the network size is large.

C. Average Energy Consumption

The average energy consumption for each WuC transmission (successful/unsuccessful) with various network sizes is illustrated in Fig. 9 for these four protocols. As shown in the figure, the energy cost of all three proposed protocols is lower than that of Cor-WuR. This is obvious because a node needs to consume nearly an equal amount of energy for a lost or a successful WuC transmission when Cor-WuR is employed. On the other hand, a node that operates on one of our protocols consumes lower energy for the lost WuC transmission since it simply discards data packet after performing CCA or BO plus CCA when the attempt threshold has been reached. Among these three protocols, CCA-WuR consumes lowest energy since it performs merely CCA prior to WuC transmissions.

D. Tradeoff of Higher Bit Rate and Short Address

We now demonstrate the benefits of applying a shorter address and a higher data rate for WuC, the two techniques presented in Section VII. Although the numerical results are obtained based on an 8-bit address and a 5460 kbps WuC transmission data rate, these parameters are application dependent and are re-configurable.

Fig. 10 illustrates the effects of these two techniques on the performance the three proposed protocols in terms of WuC loss probability, average packet delay and average energy consumption respectively. From Fig. 10(a), it is evident that the WuC loss probability decreases with a higher WuC transmission data rate. This is because the WuC duration becomes shorter with a higher data rate. Similarly, the performance with an 8-bit address is better than that of a 16-bit address for all three protocols since shorter time is needed to transmit a WuC with a smaller frame size, leading to a lower P_L.

The effects on average packet delay are presented in Fig. 10(b). All the proposed protocols exhibit similar behavior, i.e., T_S with a higher data rate (at 5460 bps), is shorter than the one that is obtained from its counterpart (at 2730 bps). When a shorter address is adopted, the average packet delay of all the protocols further reduces. For the delay behavior with respect to traffic load or network size, the same trend as shown in Fig. 7 has been observed. That is, the average packet delay increases with a larger network size for CSMA-WuR and ADP-WuR but decreases for CCA-WuR. The reason is the same as explained in Subsection VIII.B.

As can be observed in Fig. 10(c), lower energy consumption is achieved for all three protocols with a higher data rate and a shorter address length. This is because the higher the data rate, the shorter the WuC duration. So does the effect from a shorter WuC address. With a shorter WuC duration, lower

(a) WuC loss probability.

(b) Average packet delay.
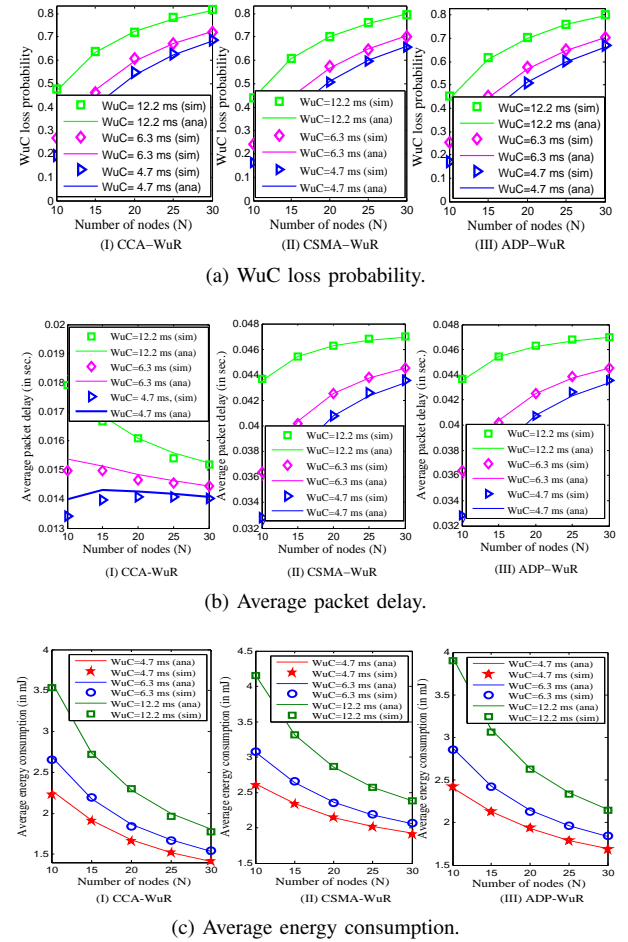

(c) Average energy consumption.

Fig. 10. Effects of data rate and address length for WuC transmissions.

energy is consumed. Since in our protocols a discarded packet consumes lower energy than a successfully transmitted packet does and the number of discarded packets increases with a larger N, the downward trend of E with N is self-evident.

Furthermore, it is expected that the performance of all these protocols deteriorates under an error-prone channel. However, the performance curves will still exhibit a similar trend as shown above under an error-free channel assumption since WuC losses due to protocol behavior is statistically independent of losses caused by channel impairments [23]. Accordingly, we can claim that Cor-WuR will reach the non-deliverable stage for packet transmissions earlier due to both channel failures and transmission collisions in an error-prone channel, whereas our proposed protocols would still be operational over a wider range of network size and traffic load conditions.

E. Performance Comparison with BoWuR

In this subsection, we compare the performance of our proposed protocols with BoWuR [17], which is a representative WuR protocol based on CSMA/CA. The BoWuR protocol performs CCA first and then BO (i.e., as in IEEE 802.11 DCF but prior to WuC transmissions). BoWuR assumes that all nodes in a network are synchronized, and can freeze their BO and defer transmissions upon overhearing a WuC
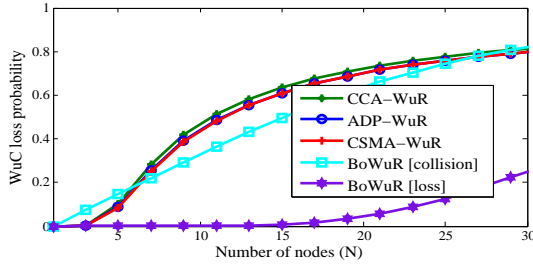
Fig. 11. WuC loss probability comparison of BoWuR, CCA-WuR, CSMA-WuR, and ADP-WuR with the WuC duration as 12.2 ms.
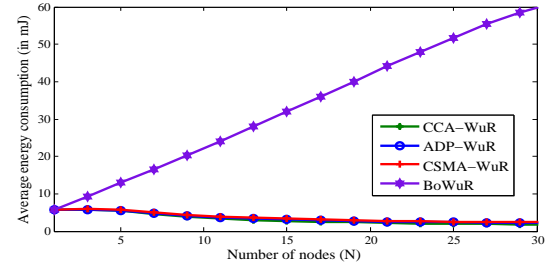

Fig. 12. Average energy consumption comparison of BoWuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

Fig. 13. Average successful packet delay comparison of BoWuR, CCA-WuR, CSMA-WuR, and ADP-WuR when the WuC duration is 12.2 ms.

transmission. It allows a node to retransmit a WuC if a collision happened in the previous transmission attempt. A node discards a WuC if the retransmission limit attempt is exceeded. The performance of BoWuR is evaluated under saturated traffic in [17]. To make a fair comparison, we have studied and analyzed the performance of BoWuR versus our protocols under unsaturated traffic condition based on [23]. For the performance comparison results presented below, we considered $N \in \{1, 2, ..., 30\}$ and the remaining parameters are configured according to Table III.

The performance of the WuC loss probability as the number of nodes varies is shown in Fig. 11. The WuC loss probabilities for these four protocols, i.e., BoWuR (marked as BoWuR [loss] in the figure), CCA-WuR, CSMA-WuR and ADP-WuR, are calculated based on the discarded packets when the attempt limit is exceeded. From the figure, it is observed that BoWuR outperforms the proposed protocols with respect to this probability. *This benefit is however achieved at an assumption on network synchronization which is not realistic for on-demand based WuR operations.* In the same figure, we show another curve for BoWuR (marked as BoWuR [collision]), illustrating the collision probability for BoWuR which will be used for delay and energy comparison in the following paragraphs.

The average energy consumption comparison of these four protocols is illustrated in Fig. 12. As expected, the proposed protocols require lower energy than BoWuR under all configured network sizes. This is because a BoWuR node still retransmits its WuC even if the previous WuC transmission is collided. With a higher collision probability (the BoWuR [collision] curve shown in Fig. 11) at a larger network size, the number of retransmissions for a successful WuC transmission will be higher, leading to higher energy consumption. On the contrary, the proposed protocols perform CCA after BO to check the channel occupancy status and continue the BO procedure if the channel is not idle, i.e., the energy needed for WuC transmissions is saved. The same procedure will be repeated until a WuC transmission is successful or the attempt limit has been reached. Furthermore, Fig. 13 presents a comparison of the average packet delay for successfully transmitted packets among BoWuR and our proposed protocols. It is observed that the average successful packet delay of all the proposed protocols is shorter than that of BoWuR for a large network. This is because a BoWuR node freezes its BO counter if it does not win the channel access competition and it has to retransmit the packet if the previous WuC transmission
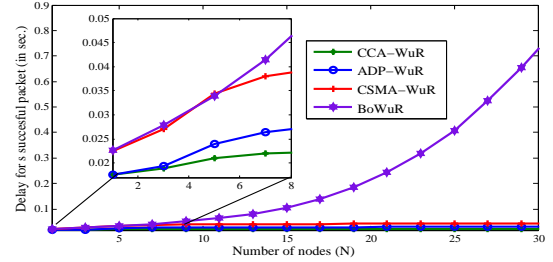
collides. The delay of BoWuR increases significantly with the number of nodes since the WuC collision probability (the BoWuR [collision] curve shown in Fig. 11) grows with a larger network size.

IX. CONCLUSIONS

In this paper, we proposed three WuR protocols known as CCA-WuR, CSMA-WuR, and ADP-WuR respectively with a focus on collision avoidance of WuCs and applying WuR to various traffic conditions. An M/G/1/2 model is developed to evaluate the performance of the proposed protocols as well as of a benchmark protocol, Cor-WuR. The obtained analytical and simulation results coincide with each other and demonstrate that all three proposed protocols outperform Cor-WuR in terms of WuC loss probability and average energy consumptions, at the cost of a longer packet delay. Among these three protocols, CCA-WuR suits best for applications where both short delay and high energy-efficient communication are required while a moderate packet loss probability is tolerable. Meanwhile, CSMA-WuR and ADP-WuR are preferable when packet delivery reliability is of more importance. Between CSMA-WuR and ADP-WuR, the latter one performs better in terms of packet delay and energy consumption. As our future work, we will further study quantitatively the impact of error rate in error-prone channels on the performance of the studied protocols, investigate the feasibility of using pseudo-orthogonal sequences for wake-up signal generation, and implement the proposed protocols in a test-bed for real-life experiment based performance evaluation. With such implementations and deployments, the applicability of WuR-based WSNs/IoT networks to industrial as well as other environments is expected to boom in years to come.

REFERENCES

[1] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks," *in Proc. IEEE INFOCOM, 2011*, pp. 1305-1313.

[2] S. Yanjun, O. Gurewitz, and D. B. Johnson, "RI-MAC: A receiver initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," *in Proc. SenSys, 2008*, pp. 1-14.

[3] G. U. Gamm, M. Sippel, M. Kostic and L. M. Reindl, "Low power wake-up receiver for wireless sensor nodes," *in Proc. ISSNIP, 2010*, pp. 121-126.

[4] G. U. Gamm and L. M. Reindl, "Range extension for wireless wake-up receivers," *in Proc. IEEE SSD, 2012*, pp. 1-4.

[5] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674-687, Apr. 2016.

[6] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," *in Proc. IEEE INFOCOM, 2015*, pp. 522-530.

[7] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. Popovici, and L. Benini, "Design, implementation, and performance evaluation of a flexible low-latency nanowatt wake-up radio receiver," *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 633-644, Apr. 2016.

[8] F. A. Aoudia, M. Gautier, M. Magno, O. Berder, and L. Benini, "Analytical and experimental evaluation of wake-up receivers based protocols," *in Proc. IEEE GLOBECOM, 2016*, pp. 1-7.

[9] M. Zhang, D. Ghose, and F. Y. Li, "Does wake-up radio always consume lower energy than duty-cycled protocols?" *in Proc. IEEE VTC-Fall, 2017*, pp. 1-5.

[10] V. Jelicic, M. Magno, D. Brunelli, V. Bilas, and L. Benini, "Benefits of wake-up radio in energy-efficient multimodal surveillance wireless sensor network," *IEEE Sensors J.*, vol. 14, no. 9, pp. 3210-3220, Sep. 2014.

[11] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Performance evaluation and comparative analysis of subcarrier modulation wake-up radio systems for energy-efficient wireless sensor networks," *Sensors*, vol. 14, no. 1, pp. 22-51, 2014.

[12] T. O. Kim, J. S. Park, H. J. Chong, K. J. Kim, and B. D. Choi, "Performance analysis of IEEE 802.15.4 non-beacon mode with the unslotted CSMA/CA," *IEEE Commun. Lett.*, vol. 12, no. 4, pp. 238-240, Apr. 2008.

[13] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535-547, Mar. 2000.

[14] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 distributed coordination function in nonsaturated heterogeneous conditions," *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 159-172, Feb. 2007.

[15] F. Z. Djiroun and D. Djenouri, "MAC protocols with wake-up radio for wireless sensor networks: A review," *IEEE Commun. Surveys & Tuts.*, vol. 19, no. 1, pp. 587-618, Q1, 2017.

[16] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra low power wake-up radios: A hardware and networking survey," *IEEE Commun. Surveys & Tuts.*, vol. 19, no. 4, pp. 2117-2157, Q4, 2017.

[17] D. Ghose and F. Y. Li, "Enabling backoff for SCM wake-up radio: protocol and modeling," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1031-1034, May 2017.

[18] S. H. Lee, Y. S. Bae, and L. Choi, "On-demand radio wave sensor for wireless sensor networks: Towards a zero idle listening and zero sleep delay MAC protocol," *in Proc. IEEE GLOBECOM, 2012*, pp. 560-566.

[19] P. Sthapit and J.-Y. Pyun, "Effects of radio triggered sensor MAC protocol over wireless sensor network," *in Proc. IEEE CIT, 2011*, pp. 546-551.

[20] F. A. Aoudia, M. Gautier, and O. Berder, "OPWUM: Opportunistic MAC protocol leveraging wake-up receivers in WSNs," *Journal of Sensors*, vol. 2016, Article ID 6263719, 2016.

[21] J. M. Lebreton, S. Kandukuri, N. Murad, and R. Lorion, "An energy-efficient duty-cycled wake-up radio protocol for avoiding overhearing in wireless sensor networks," *Wireless Sensor Network*, vol. 8, pp. 176-190, Aug. 2016.

[22] K. R. Chowdhury, N. Nandiraju, D. Cavalcanti, and D. P. Agrawal, "CMAC-A multi-channel energy efficient MAC for wireless sensor networks," *in Proc. IEEE WCNC, 2006*, pp. 1172-1177.

[23] F. Daneshgaran, M. Laddomada, F. Mesiti, and M. Mondin, "Unsaturated throughput analysis of IEEE 802.11 in presence of non ideal transmission channel and capture effects," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1276-1286, April 2008.

[24] IEEE Standard, 802.15.4e, "Part 15.4: Low-Rate Wireless Area Network (LR-WPANs) Amendment 1: MAC sublayer", New York: IEEE Standard for Information Technology, 2012.

[25] Texas Instruments, *CC1101 low-power sub-1 GHz RF transceiver datasheet* , USA, Nov. 2013.

[26] NXP Semiconductors, *JN-AN-1001 JN516x Power Consumption*, Application Note, Eindhoven, The Netherlands, Jun. 2016.

[27] *AMS AS3932 Data sheet*, Austria, Mar. 2015.

Debasish Ghose received the B.Tech. degree in Electronics and Telecommunication Engineering from KIIT University, India, in 2008 and the M.Sc. degree in Communication Engineering from National Taipei University, Taiwan, in 2014. From 2008 to 2011, he worked in Tata Consultancy Services Limited, India, as a system engineer. He is currently working toward the Ph.D. degree in Information and Communication Technology at the University of Agder, Norway. His doctoral research is focused on protocol design, modeling and performance evaluation of wake-up radio networks. His other research interests include WiFi, LiFi, WSN, and IoT networks.

Frank Y. Li received the Ph.D. degree from the Department of Telematics, Norwegian University of Science and Technology (NTNU), Trondheim, Norway. He worked as a Senior Researcher at UniK-University Graduate Center (now Department of Technology Systems), University of Oslo, Norway before joining the Department of Information and Communication Technology, University of Agder (UiA), Agder, Norway, in August 2007 as an Associate Professor and then a Full Professor. During the past few years, he has been an active participant in several Norwegian and EU research projects. He is listed as a Lead Scientist by the European Commission DG RTD Unit A.03-Evaluation and Monitoring of Programmes in Nov. 2007. Dr. Li's research interests include MAC mechanisms and routing protocols in 5G mobile systems and wireless networks and the Internet of Things; mesh and ad hoc networks; wireless sensor networks; D2D communication; cooperative communication; cognitive radio networks; green wireless communications; reliability in wireless networks; QoS, resource management and traffic engineering in wired and wireless IP-based networks; analysis, simulation and performance evaluation of communication protocols and networks.

Vicent Pla received the Telecommunication Engineering (B.E. & M.E.) and Ph.D. degrees from the Universitat Politècnica de València (UPV), Spain, in 1997 and 2005, respectively, and the B.Sc. in Mathematics from the Universidad Nacional de Educación a Distancia (UNED), Spain, in 2015. In 1999, he joined the Department of Communications at the UPV, where he is currently a Professor. His research interests lie primarily in the area of modeling and performance analysis of communication networks. During the past few years, most of his research activity has focused on traffic and resource management in wireless networks. In these areas he has published numerous papers in refereed journals and conference proceedings, and has been an active participant in several research projects.