

# MAC Layer Throughput Estimation in Impulse-Radio UWB Networks

Ioannis Broustis, *Member, IEEE*, Angelos Vlavianos, *Student Member, IEEE*, Prashant Krishnamurthy, *Member, IEEE*, and Srikanth V. Krishnamurthy, *Member, IEEE*

**Abstract**—The inherent channel characteristics of impulse-based UWB networks affect the MAC layer performance significantly. Most previous studies on evaluating MAC protocols are based on prolonged simulations and do not account for the multiple-access interference due to multipath delay spread. In this work, we develop CTU, an analytical framework for Capturing the Throughput dependencies in UWB networks, while taking into account the PHY layer effects. The key attributes of CTU are: (a) It is modular; it can be easily modified to provide a basis for evaluating a wide range of MAC protocols for impulse-based UWB networks. The only requirements are that the MAC protocol under study be based on time-hopping and the modulation scheme be pulse position modulation; these are common design decisions in UWB networks. (b) It considers the channel characteristics in addition to MAC layer effects; CTU correlates probabilistically the multipath delay profile of the channel with the packet error rate. We employ CTU to evaluate the performance of different generic medium access procedure. We compare the results with those from extensive simulations and show the high accuracy of CTU. We use CTU to assess the impact of various system parameters on the MAC layer performance; we make several interesting observations that are discussed in depth.

**Index Terms**—Ultra-Wide Band (UWB), Wireless Communications, Multipath Delay Spread, Modulation.



## 1 INTRODUCTION

The medium access performance of multi-hop impulse-based UWB networks is significantly affected by the PHY layer characteristics, and in particular by the *multipath delay spread* [1]. Due to reflections from various objects, multiple copies (rays) of the same transmitted signal appear at the receiver; each copy has a different amplitude, phase and delay. The copies interfere with later pulse transmissions and this significantly affects the long-term network throughput. Clearly, the extent to which the multipath delay spread affects the performance depends on the pulse rate at the PHY layer, the topology, as well as the ability of the receiver's hardware to compensate with the ray reception, as we discuss later. Therefore, when evaluating MAC protocols designed for UWB networks, it is critical that one accounts for these effects. Most previous studies on UWB, however, are limited to either (i) possibly time-intensive simulation-based MAC protocol evaluations (most of such efforts do not account for multipath delay spread effects), or (ii) PHY layer analyses to model the behavior of a single link in the presence of delay spread. A cross-layer analytical framework that quickly and accurately quantifies the impact of the UWB PHY layer attributes on MAC layer mechanisms, can be of great help throughout the protocol design process.

In this paper, we develop an analytical framework to compute the average saturation throughput and evaluate its dependencies on PHY layer effects, in multi-hop UWB networks. The saturation

throughput is defined to be the throughput when the nodes in the network always have packets to send, i.e., their outgoing queues are always non-empty. We call our framework “CTU” for short (for Capturing Throughput dependencies in UWB networks). The key attributes of CTU are: (a) it is modular and therefore flexible, and (b) it captures both physical layer characteristics and MAC layer effects. CTU computes the probability of packet loss due to (i) MAC layer collisions and (ii) multiple access interference (MAI) due to delay spread effects. It then combines them to compute the saturation throughput. The immediate benefit from our analysis is that it obviates the need for *repeated* simulations. We elaborate on the attributes of CTU below:

**CTU is modular and hence applicable to a wide set of MAC protocols.** The part of the CTU framework that computes packet losses due to MAI is generic and can be used in conjunction with any MAC layer protocol. The only requirements are that the protocol be based on time-hopping (TH), a multiple access procedure, and that the underlying modulation scheme be Binary Pulse Position Modulation (BPPM); this is common to most PHY/MAC layers designed for UWB networks. The building block for CTU is a basic MAC layer procedure that is likely to be the foundation for more complex TH-based MAC layer protocols.

**CTU accounts for cross layer dependencies.** Multiple-access interference in UWB networks not only depends on the extent of delay spread but also on the access procedure defined by the MAC protocol. CTU provides a seamless methodology of integrating the access procedure with the effects of delay spread; in other words, it captures the cross layer dependencies between the PHY and the MAC layers. In particular, CTU incorporates the impact of delay spread on the multiple access performance based on a modified version of the Saleh-Valenzuela (SV) model that has been adopted by the IEEE 802.15.3a task group [2], (discussed in more detail in section 2).

**Evaluation of CTU.** We apply CTU on a simple MAC protocol, and we compute the per node throughput. We validate the accuracy of our analytical results, by comparing them with

- 
- I. Broustis is currently with Alcatel-Lucent, NJ.  
E-mail: ioannis.broustis@alcatel-lucent.com
  - A. Vlavianos is with 7Layers Co., Athens.  
E-mail: aggelos@7layers.org
  - S. V. Krishnamurthy is with University of California, Riverside.  
E-mail: krish@cs.ucr.edu
  - P. Krishnamurthy is with University of Pittsburgh.  
E-mail: prashant@mail.sis.pitt.edu

This work is supported in part by the NSF CAREER Grant No. 0237920 and the NSF NRT grant No. 0335302.

results from extensive simulation experiments that we perform. We also find that while obtaining simulation results takes days, generating results with CTU requires less than one hour.

**Using CTU to assess system performance.** With CTU, we examine the impact of various system parameters on the saturation throughput. We discuss in detail, many interesting trends that are observed. We believe that CTU can help designers and implementers choose UWB MAC layer system parameters appropriately during network design.

The rest of the paper is organized as follows. In section 2 we provide background on the UWB PHY layer and discuss related work. In section 3, we describe our analytical framework in detail. In section 4, we compare the simulated behavior of the candidate MAC protocol with the behavior observed with CTU. We also describe the impact of various system parameters on network throughput under saturation conditions, report the trends that we observe and the interpretations thereof. In section 5 we demonstrate how CTU can be adjusted to facilitate different MAC protocols for UWB. In section 6 we elaborate on some of our assumptions and we discuss the limitations of CTU. Our conclusions form section 7.

## 2 BACKGROUND AND RELATED WORK

In this section, we briefly discuss the UWB PHY layer, the SV model and related previous studies [3], [4], [5].

### 2.1 Time-hopped Binary-PPM

Time hopping has been used in previous approaches for sharing a single impulse-based UWB frequency band among multiple users [6]. In mobile ad hoc networks, imposing fixed TDMA-like schedules is difficult due to the fact that nodes could be mobile. A completely random access scheme is (a) unlikely to provide high throughput and (b) requires nodes to acquire synchronization at arbitrary unpredictable time instants. Time-hopping is a form of spread spectrum communications specifically designed for impulse based systems. In single-band approaches, nodes transmit as per pseudo-random time-schedules; the pseudo-random nature of time-hopping provides a reasonable level of robustness to collisions. To the best of our knowledge, all protocols designed thus far, for impulse based UWB ad hoc networks, use time-hopping as the basic means of providing multiple-access. In time hopping, a fixed number of chip-times are aggregated to form a **sequence frame**. The duration of each sequence frame is  $T_f$  and thus the number of chip-times per sequence frame is  $T_f/T_c$ . Binary PPM (BPPM) signals are comprised of pulses; each transmitter sends a pulse in only one of the chip-times in each sequence frame. The specific chip-time is determined by the node's time hopping sequence (THS), generated as per a pseudo-random number (PN) code. Each pulse represents an encoded bit, which can be a '0' or a '1', as determined by the pulse position during  $T_c$ . If the pulse occupies the first slot, it represents a "0", otherwise a "1". A Viterbi decoder is usually embedded at the receiver [7] for the soft-decision decoding of the received information.

Time hopping sequences may be either sender-based or receiver-based. In receiver-based time hopping, a receiver expects to receive a pulse only in one of the chip-times in a sequence frame. In the sender-based case, the transmitter sends pulses based on its own THS. The sender-based strategy is robust; however

the receiver has to be synchronized with all of its potential transmitters. The receiver-based approach is much simpler; however one could encounter collisions between the pulses from different transmitters, directed towards the same receiver. Protocols could use both approaches, as in the one that we describe in the following section.

Note that it is extremely difficult to guarantee that time hopping sequences of nodes are orthogonal to each other. To satisfy this requirement, the number of chips within a sequence frame have to be *larger* than the number of nodes in the network, i.e., each node will need to have a dedicated chip-time for duration. As we show in section 4, this can result in extremely long sequence frames and as a consequence to longer delays, especially at low loads. Continuing the above discussion, the average spacing between successive transmissions as per the THS will have an effect on the achieved performance, as we analytically show later. With shorter spacing between the time-hops, the pulses could be sent at a faster rate; however, there is a higher possibility of collisions. With longer spacing, the possibility of collisions is reduced; however, large delays could be incurred. Note also that even if time hopping sequences are orthogonal, parallel transmissions can still cause interference to each other. This is because: (a) Due to multipath delay spread, rays may appear in neighbor successive chips, thereby distorting pulse receptions that belong to orthogonal hopping sequences (we analyze the impact of this effect in Section 3). (b) Nodes may transmit in an asynchronous manner, and this can cause pulse collisions. Although this is possible, in this paper we assume that nodes transmit in a synchronous manner.

### 2.2 The modified SV model

UWB transmissions experience *multi-path delay spread*. A transmitted UWB pulse, radiated using an isotropic antenna, will take multiple paths (as a consequence of reflections from various objects) and results in multiple time-shifted copies at the receiver [8]. Each received copy (called ray) may have a different amplitude, phase and delay. Beyond a certain delay threshold, called the delay spread of the channel, the signal amplitudes may be considered negligible. For indoor environments, measurements have shown that the delay spread is of the order of tens of nanoseconds. If the time-spacing between the UWB pulses is smaller than the delay spread of the channel, copies of a transmitted encoded bit interfere with the subsequent encoded bits. This is called *inter-symbol interference* or ISI for short. Various methods have been proposed in order to combat ISI, such as equalization [7] and Maximum Likelihood Sequence Detection (MLSD) [9]. Equalization methods could be used to alleviate the effects of ISI [7] due to signals emitted from the associated transmitter. A drawback with equalizers is that they cannot mitigate interference due to rays produced by other neighbor transmitters that are active simultaneously. Our study captures these effects; as we explain in the following section, we assume that each receiver is equipped with an equalizer, and we capture the impact of ISI due to parallel pulse transmissions from different devices. Furthermore, MLSD is a decoding mechanism for trellis codes [10] that operate over linear ISI channels [11]. Traditional MLSD detectors are quite complex and therefore difficult to implement in practice [12]. Thus, a combination of reduced-complexity MLSD algorithms and equalizers is a potential strategy that can also be practically adopted.

The SV channel model characterizes the arrival of rays at the receiver [4]. As per this model, rays arrive at the receiver in clusters. Within each cluster, rays appear in time as per the Poisson( $\lambda$ ) arrival process:

$$p(\tau_{\nu,l}|\tau_{(\nu-1),l}) = \lambda e^{-\lambda(\tau_{\nu,l}-\tau_{(\nu-1),l})}, \quad (1)$$

where  $\tau_{\nu,l}$  is the delay of the  $\nu$ <sup>th</sup> ray, relative to the  $l$ <sup>th</sup> cluster arrival time,  $T_l$  and the previous,  $(\nu-1)$ <sup>st</sup> ray. Similarly, clusters also appear as per a second Poisson( $\Lambda$ ) process:

$$p(T_l|T_{l-1}) = \Lambda e^{-\Lambda(T_l-T_{l-1})}. \quad (2)$$

In each cluster, the power in the component rays, decays as per an exponential-decay profile. In addition, the model determines the discrete time impulse response of the signal to be:

$$h(t) = \sum_{l=0}^L \sum_{\nu=0}^V \alpha_{\nu,l} \delta(t - T_l - \tau_{\nu,l}), \quad (3)$$

with  $\alpha_{\nu,l}$  being the multipath gain coefficient. The primary version of the SV model employs a Rayleigh distribution for the multipath gain magnitude. In particular,  $\alpha_{\nu,l} = p_{\nu,l} \beta_{\nu,l}$ , where  $p_{\nu,l}$  is equiprobably  $\pm 1$  and  $10 \log_{10} |\beta_{\nu,l}|^2 \propto Normal(\mu_{\nu,l}, \sigma^2)$ . Moreover,  $\mu_{\nu,l}$  is given by:

$$\mu_{\nu,l} = \frac{10 \ln(P_m) - 10T_l/\Gamma - 10\tau_{\nu,l}/\gamma}{\ln 10} - \frac{\sigma^2 \ln 10}{20} \quad (4)$$

where  $P_m$  is the mean power of the first ray of the first cluster, and  $\Gamma$ ,  $\gamma$  are the cluster and ray decay-factors, respectively. Experimental data, using Kolmogorov-Smirnov testing with a 1% significance level suggested that the lognormal amplitude distribution seems to fit the profile better than the Rayleigh distribution [5]. Thus, in this paper we adopt the lognormal distribution for the channel coefficients; besides its accuracy, for the purposes of our study this model is also easier to approach analytically than other models. We will revisit this model later, in section 3.

## 2.3 Related studies

There are various studies on proposing analytical models for the evaluation of MAC protocols, designing and evaluating MAC protocols through simulations, as well as presenting other channel models. We discuss these papers in what follows.

**Analytical studies on modeling the performance of IR-UWB MAC protocols:** Our work is one of the few efforts to propose an analytical framework for MAC protocol evaluation in pulse-based UWB. An interesting study appears in [13], [14], where Merz et al. focus on evaluating how achieving synchronization through common or private acquisition preambles affects the network throughput. They derive an analytical model (based on a decoupling approximation where all packet sources are assumed to have an identical and independent behavior) to compute the saturation throughput with the two approaches. Their model can be applied with different appropriate MAC protocols and acquisition algorithms. They show that private acquisition offers higher throughput benefits, especially in scenarios with many parallel transmissions. In addition, they compare their analytical results with those from their ns-2 simulations, while they also consider TCP traffic and large topologies. Giancola and Di Benedetto in [15] present a novel method for estimating the BER (Bit Error Rate) in impulse-based UWB networks with multi-user interference. They consider the potential pulse collisions occurring

at reference receivers and propose an analytical expression for the average BER over flat AWGN channels. They derive the probability of error taking into account collisions between pulses of different transmissions. Although they consider the amplitude of the different received pulses, they do not include any multipath delay profile and therefore they ignore the effects of the delay spread. Guvenç et al. in [16] investigate the effects of adapting various UWB parameters for multiple access, using a Gaussian approximation method to model the multiple access interference. The main parameters of interest are the number of pulses per bit and the number of time slots per time frame. They are interested in observing how the adaptation of these parameters per user (considering the channel quality observed by each of them) affects the system behavior. However, in contrast to our work, they do not assess the effects of those parameters on system throughput; they do not provide an analysis for estimating the per-node throughput based on the probability of successful packet reception. More than that, they do not show how the UWB frame length or the pulse repetition affect the probability of correct bit reception or the probability of packet corruption.

**Studies on new MAC protocols for impulse-based UWB:** In [1], [17] we propose multi-band MAC protocols to alleviate the impact of both THS overlaps and multipath delay spread. Le Boudec et al. [6], [18] propose a scheme that uses dynamic channel coding. The transmitter dynamically varies the code rate upon receiving feedback from the receiver with regard to channel conditions. The scheme uses two types of THS: a receiver-based THS and an invitation-based THS. This invitation-based THS is unique for each communicating pair. After the successful transmission of a request using the receiver based THS, the pair switches to a *unique* invitation-based THS and uses this THS for the duration of the session. Along similar lines in [19] an enhanced joint PHY/MAC architecture and a private MAC are presented for very low power UWB, where nodes listen to up to 3 hopping sequences at the same time; they always listen on a common broadcast THS and their own when idle. The receiver-based THS approach is followed for transmissions. U-MAC is proposed in [20]. In order to optimize the global network performance, U-MAC assigns rate and power values to nodes through state declarations, which are embedded into periodic *hello* message transmissions. The periodicity of hello messages considers the network stability in order to avoid unnecessary frequent reports and reduce the control overhead. In addition, the radius around each receiver is adjusted in order to provide fairness between all sessions to the receiver. In [21] the authors describe theoretical and practical approaches towards the development of a THS based MAC protocol for radio resource sharing in UWB ad hoc networks. Benedetto et al. in [22] present UWB<sup>2</sup>, a pure Aloha-based MAC protocol for low bit-rate UWB networks. UWB<sup>2</sup> relies on the observation that a sufficient number of pulses in the synchronization sequence may lead to successful synchronization detection with high probability, even in the presence of multi-user interference. UWB<sup>2</sup> provides a hybrid scheme for THS assignment, which combines a common control channel (through a common TH code) with dedicated data channels (through certain transmitter-receiver pair TH code assignments). Although these studies build MAC protocols around the notion of pulse-level collisions, they consider simple channel and interference models, as well as low data rate scenarios where ISI is not prominent. Ding et al study issues related to channel

acquisition [23]. They conclude that existing MAC solutions are unsuitable for UWB networks. Specifically, the authors examine TDMA and CSMA/CA approaches, which have been successful in other environments. In [24], all nodes share a THS and the receiver broadcasts an invitation, as per this sequence. Potential transmitters compete during a contention period, to lock on to the receiver. In [25] the authors propose a full-duplex access scheme for impulse-based UWB networks. The scheme takes advantage of the low duty cycle to maintain physical links among two nodes for the lifetime of their logical link, thereby removing the requirement that the sender and receiver resynchronize for every packet to be exchanged. A theoretical treatment on optimal routing, scheduling and power control appears in [26]. The authors show that (a) the design of the optimal MAC is independent of the choice of the routing protocol and (b) a minimum energy route is preferable to establishing long hops or invoking direct transmissions. A MAC protocol for single-transceiver UWB ad hoc networks based on the use of busy signals is proposed in [27]. The key objective of the MAC protocol is to facilitate the detection of collision of UWB signals by using UWB pulses. However, the protocol is not designed with the intent of maximally exploiting the available spectrum in the presence of delay spread.

**Studies on the Impact of Delay Spread:** In [28], the authors compute the conditional bit error rate for TH-BPSK (Time-Hopped Binary Phase Shift Keying) UWB, in a multi-user, multipath-channel environment. However, unlike with our work, the authors assume that a pulse from an interferer will collide with at most one pulse from the user of interest. In addition, they do not study the cross layer dependencies between the MAC and PHY layers. Gubner and Hao, in [29], derive an analytical formula for the average bit error probability at the PHY layer, considering the IEEE 802.15.3a model. They observe that, if one were to consider an observation window that accounted for 99% of the interference energy due to pulse copies, the performance over a link would be identical to a case where one considers the delay spread to be infinite. In other words, they show that there is a maximum duration (maximum value of the delay spread) over which pulse copies will need to be considered. The authors however, do not consider multi-access interference or MAC level characteristics. In [30] we perform a simulation study of the impact of delay spread on the throughput in UWB networks with a simple model; we do not consider any ray fading model (as with the SV model).

**Channel Models:** There exists a lot of work on modeling the multipath delay spread profile. Foerster [5] proposes a variation of the SV model that adopts the Nakagami fading distribution. In [31] Cassioli et al. consider only one cluster with an exponential decay of the ray powers. Ghassemzadeh et al. [32] modify the single-cluster model of [31] to include a noise-like variation with lognormal statistics. Greenstein et al., in [33], present an interesting comparison of these models.

### 3 OUR ANALYTICAL FRAMEWORK

In this section we present CTU, a framework for estimating the MAC layer performance, in pulse-based UWB networks. In order to make the description easy to understand, we first highlight the main idea and discuss the assumptions that we make throughout the design of our framework. Moreover, we explain the operations of a simple MAC procedure, which we use as a benchmark to progressively build CTU and justify our design decisions.

#### 3.1 The core idea with CTU

Our primary task is to capture the impact of the multipath delay spread on the MAC layer performance, taking into account (a) the inherent properties of BPPM-enabled links, and (b) the MAC protocol scheduling aspects. In particular, we utilize the SV model, to approximate the level of interference within a chip time, due to ray arrivals. This enables us to compute the probability of successful decoding of a bit, and progressively a MAC-layer packet.

The packet success depends on the set of actions dictated by the MAC layer protocol that is considered by the designer. In particular, a packet transmission may fail due to any of the following two reasons:

1) The packet collides because more than one senders transmit simultaneously towards the same receiver. Here each transmission fails because the same THS is being used. We denote the probability of such a collision as  $P_{MAC}$ .

2) The packet does not collide in the above case; however, it is partially corrupted, due to high multi-access interference levels (MAI). This partial corruption may occur because: (a) individual pulse transmissions may collide due to partial THS overlaps, and (b) there is a high interference energy due to multipath components. We argue that the impact due to (a) is negligible (later in the section) and denote the probability of such a partial corruption due to (b) by  $P_{MAI}$ . In particular:

$$P_{failure} = P_{MAC} + (1 - P_{MAC}) \cdot P_{MAI} \quad (5)$$

The goal of CTU is to compute the per node throughput by utilizing the value of  $P_{failure}$ . The above probabilities are interdependent, since the computation of  $P_{MAC}$  utilizes the value of  $P_{failure}$ , as we demonstrate later in this section. *Therefore, the calculation of  $P_{failure}$  is an iterative process.* Before proceeding with the presentation of our design, we provide a detailed list with our assumptions in what follows.

#### 3.2 Design assumptions

For analytical tractability, we make the following assumptions; we elaborate on some of them in section 6.

- 1) All nodes are independent from each other, i.e., the network is decoupled.
- 2) Nodes always have packets for transmission in their queues. In other words, we consider conditions of saturation.
- 3) Only nodes that are within the transmission range  $R$  of a receiver can interfere with packet transmissions towards that receiver. Furthermore, the transmission range is much smaller than the deployment region.
- 4) Each bit is represented by  $PB$  repetitions of a pulse transmission (repetition coding).
- 5) We assume that both the transmission and the reception of the same pulse occur during the same chip.
- 6) The duration of the multipath delay spread is fixed and equal to  $\Delta$ ; this implies that we are interested in rays for which the received power is above a certain threshold, a reasonable approach as shown in [29]. We assume that  $\Delta$  corresponds to an integer number of chips:  $\Delta = N_c \cdot T_c$ .
- 7) Receivers are equipped with an equalizer. Therefore, rays do not interfere with subsequent pulses from the *same* transmitter. In addition, we assume that bit errors are independent.

- 8) The path-loss is fixed; we assume an attenuation of 20 dB for a transmission rate of 110 Mbps, over a distance of 10m [5], and for both the given transmitter and interferers. We elaborate on this assumption in Section 6.
- 9) There is no overlap between the rays induced by a single transmitter in any slot; the received pulses/rays are not distorted. The rays are assumed to arrive within slot boundaries.
- 10) We assume randomly assigned (potentially overlapping) THSs. Therefore a given transmitter may transmit within a given chip of a frame with the same probability as within any other chip in that frame.
- 11) The number of active transmitters that can cause interference remains a constant during a packet's transmission. This number can actually increase in some cases while decrease in others; the former effect is detrimental to packet receptions while the latter aids the process. Thus, on average, one might expect the behavior to be similar to that of the assumed system.
- 12) Two or more packet transmissions sent to the same receiver at the same time with the same THS will collide.

Let us now consider a network deployment consisting of  $N$  nodes, randomly and uniformly deployed across a square region  $A$ . CTU estimates the MAC-layer network throughput, by computing the packet failure probability,  $P_{failure}$ . Throughout the rest of this section we describe the procedure of computing  $P_{failure}$ , which determines the per-node long-term throughput; for this we consider a basic MAC layer procedure as a driving example, discussed in what follows.

### 3.3 The basic MAC layer procedure

The single-band MAC layer procedure that we use for modeling purposes is a simplified version of the one in [6]. Note that there could exist more intelligent MAC protocols [34] than the one considered here. CTU can be easily tuned to characterize different impulse-based protocols (as explained later in section 6). In our considered toy MAC protocol, nodes utilize randomly overlapping THSs. The protocol operations are shown in Fig. 1.

Let us assume that Jack wants to send a data packet to his neighbor, Chloe (Fig. 2). Jack is in the *REQ* state, while Chloe is in the *IDLE* state. The following takes place (Fig. 1).

**REQ state:** Jack transmits a request (REQ) to Chloe (a randomly selected neighbor), as per Chloe's THS. He then switches to a THS that is unique to Jack and Chloe. Nodes in the REQ state are only the transmitters that send REQ messages. Their potential receivers may be in any of the other three states; however, the REQs can be received successfully only when the receiver is in the *IDLE* state.

**DATA state:** After a successful reception of the REQ, both Jack and Chloe are in the *DATA* state, wherein Chloe responds to Jack's request with a RACK message. Note that in the *DATA* state, a THS unique to Jack and Chloe, is used. Jack now knows that Chloe has accepted his request, and he proceeds with his *DATA* packet transmission. Chloe acknowledges the *DATA* reception with an ACK message to Jack. At this point, the MAC layer session between Jack and Chloe has ended. Both Jack and Chloe further transmit a short BEACON message, on their own THSs, to inform their neighbors that they are now idle. Then, both of them transit to the *IDLE* state.

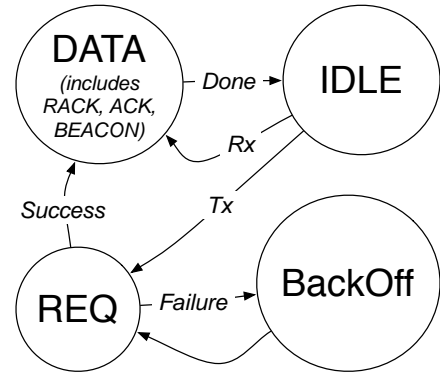


Fig. 1. MAC state diagram. The *DATA* state also includes the transmission of RACK, ACK and BEACON messages.

**IDLE state:** A node resides in the *IDLE* state for a random duration, waiting for potential REQ transmissions. If no REQ arrives within this time, the node will go to the REQ state, to initiate a new session. Note that if Chloe is in the *IDLE* state, and the random duration expires before Jack completes a REQ transmission to her, Chloe will abort this REQ reception, and will transition to the *REQ* state. Again, the model for this simple procedure can be easily replaced with other possibilities.

**BackOff State:** If Bill is transmitting a REQ to Chloe at the same time as Jack, the two REQ messages will collide<sup>1</sup> since both Jack and Bill are using Chloe's THS for their REQ transmissions. Jack and Bill will realize that a collision has taken place, since Chloe will not send a RACK back. Jack and Bill will both go to the *BackOff* state, and will initiate their random backoff timers. When a timer expires, the corresponding node will re-transmit the REQ to Chloe. With this backoff algorithm, the contention window increases exponentially from 0 to  $J$  for the first  $\xi$  REQ collisions, and remains at  $J$  for up to  $\zeta$  failures. If the REQ does not go through in  $\zeta$  consecutive attempts, Jack will abort his transmission, and will send a new REQ to another, randomly selected neighbor. Moreover, if Chloe is already in the *DATA* state with Bill, Jack will go to the *BackOff* state, until he receives a BEACON message from Chloe, notifying him that she is now idle.

With this MAC protocol a transmitter selects a neighbor as the receiver at random; i.e., we consider a scenario where an application sends packets to random receivers. This behavior is also possible when a node maintains sessions with many/all of its neighbors. Note that only REQ messages are likely to suffer MAC-level collisions (see the definition of  $P_{MAC}$  above). Indeed with REQ messages, Chloe's THS may be used by both Jack and Bill at the same time; for RACK, *DATA*, ACK and BEACON transmissions, unique THSs are used.

Using the basic MAC protocol described above, CTU computes the probability of a MAC-layer packet success, as we discuss in what follows.

1. With an IR-UWB receiver, it is possible that Chloe synchronizes with one THS, and that the other transmission creates interference. Such a behavior can be incorporated to CTU; to make the analysis more tractable we assume that two or more parallel transmissions towards the same receiver always collide.

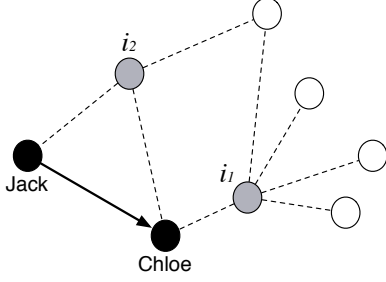


Fig. 2. Receiver's neighborhood: Jack wants to send a packet to Chloe, while there are  $i$  other neighbor transmitters (here  $i_1$  with  $l_1=5$ , and  $i_2$  with  $l_2=3$ ).

### 3.4 Constructing CTU

In order to compute  $P_{failure}$  as per Eq. (5), we need to derive  $P_{MAC}$  and  $P_{MAI}$ . We derive these probabilities in what follows.

#### 3.4.1 Computing the collision probability $P_{MAC}$

Towards calculating  $P_{MAC}$ , one needs to compute the probability of MAC collision for each of the packets (control and data) that are exchanged throughout a session between a transmitter and a receiver. In the general case,  $P_{MAC}$  will then be equal to the sum of these probabilities. As we explain below, the probability of a MAC collision is strongly dependent upon the density of the network, in terms of the number of nodes located in the vicinity of a node.

With our simplistic MAC protocol described in section 3.3,  $P_{MAC}$  can be non-zero for REQ transmissions only. Let us denote by  $P_{REQ}$ ,  $P_{BackOff}$ ,  $P_{IDLE}$  and  $P_{DATA}$ , the probabilities of a node being in the *REQ*, *BackOff*, *IDLE* and *DATA* states, respectively. The transition from one state to the other is a Markov process [35]; thus the probability  $P_{REQ}$  that a node resides in the REQ state depends on the expected time  $E_t$  in each state, i.e.,:

$$P_{REQ} = \frac{E_t[REQ]}{E_t[REQ] + E_t[BackOff] + E_t[DATA] + E_t[IDLE]}. \quad (6)$$

Let us again assume that Jack wants to send a REQ to Chloe (Fig. 2). If at least one more transmitter (besides Jack) sends a REQ to Chloe in parallel with Jack, both REQs will collide. In this case,  $P_{MAC}$  is equal to the probability that at least one other of Chloe's neighbors transmits a REQ to Chloe, at the same time as Jack. Clearly, this is heavily dependent on the *transmitter* density around Chloe. The probability that there are  $K$  nodes within Chloe's sensing vicinity, given that there are  $N$  nodes in the network, is:

$$P_{range}(K) = \binom{N}{K} \cdot \left(\frac{\pi R^2}{A}\right)^K \cdot \left(1 - \frac{\pi R^2}{A}\right)^{N-K}, \quad (7)$$

assuming a radial coverage of range  $R$ . Let Chloe have  $M$  active neighbor transmitters, including Jack. The probability  $P_{Chloe}$ , that none of the other  $M-1$  transmitters picks Chloe, equals:

$$P_{Chloe} = \sum_{i=0}^{M-1} \binom{M-1}{i} (P_{REQ})^i (1 - P_{REQ})^{M-1-i} P_{notChloe}(i),$$

where  $P_{notChloe}(i)$  is the probability that the specific  $i$  transmitters *do not* pick Chloe to send a REQ at this point in time. To compute  $P_{notChloe}(i)$ , we consider the probability that the  $\vartheta_{th}$

neighbor transmitter ( $\vartheta \in i$ ) will not pick Chloe, given that  $\vartheta$  has  $l_\vartheta$  neighbor potential receivers (Fig. 2). With this,

$$P_{notChloe}(i) = \prod_{\vartheta=0}^i \left[ \sum_{l_\vartheta=1}^{N-1} \left( \left(1 - \frac{1}{l_\vartheta}\right) \cdot P_{range}(l_\vartheta) \right) \right] \quad (8)$$

where  $P_{range}(l_\vartheta)$  is given by Eq. (7). We now define  $P_{tx} = P_{REQ} + P_{DATA}$ , where  $P_{DATA}$  is the probability that a node is in the *DATA* state, and is computed in a manner similar to the calculation of  $P_{REQ}$  in Eq. (6), with  $E_t[DATA]$  in the numerator, instead of  $E_t[REQ]$ . The probability that the REQ from Jack to Chloe does not collide with another REQ towards Chloe, is equal to  $P_{Chloe}$  times  $P_{TX}(M-1)$ , the probability that Chloe actually has  $M-1$  neighbor transmitters:

$$P_{TX}(M-1) = \sum_{n=M-1}^N \binom{N}{M-1} P_{tx}^{M-1} (1 - P_{tx})^{n-(M-1)} P_{range}(n). \quad (9)$$

As a consequence, for all possible values of  $M$ :

$$P_{MAC} = 1 - \sum_{M=2}^{N-1} P_{Chloe} \cdot P_{TX}(M-1). \quad (10)$$

We now have an expression for  $P_{MAC}$ , which depends on the expected time of residing in each state, as per Eq. (6). We still need to calculate the constituent terms of Eq. (6).  $E_t[DATA]$  is easy to compute; it corresponds to the expected number of chips needed for the *DATA*, *RACK*, *ACK* and *BEACON* transmissions. This is equivalent to:

$$E_t[DATA] = N_f \cdot P_B \cdot (DATA + RACK + ACK + BEACON) \quad (11)$$

in number of chips<sup>2</sup>.

We derive  $E_t[REQ]$ , as follows. If Jack's REQ transmission fails, for each subsequent REQ retransmission for that particular request, Jack will reside in the *REQ* state. The probability of having exactly  $r$  successive REQ failures is:

$$P_r^{REQ} failures = (P_{failure}^{REQ})^r (1 - P_{failure}^{REQ}).$$

We wish to point out here that  $P_{failure}^{REQ}$  will ultimately be computed *iteratively* by Eq. (5). From the above:

$$E_t[REQ] = \sum_{r=0}^{\zeta} (r+1) N_f \cdot P_B \cdot REQ \cdot P_r^{REQ} failures. \quad (12)$$

With regards to computing  $E_t[BackOff]$ , the backoff contention window for the first  $\xi$  attempts follows the discrete uniform distribution in the value field  $(0, 2^x - 1)$ , with  $x = 3, 4, \dots, 8$ . The average value for the contention window is  $\frac{(2^x - 1) + 1}{2} = \frac{2^x}{2}$ . Hence:

$$E_t[r \text{ backoff } fs] = \begin{cases} \sum_{b=1}^r \frac{2^{b+2}}{2}, & r \leq \xi \\ \sum_{b=1}^6 \frac{2^{b+2}}{2} + \sum_{b=7}^r \frac{2^8}{2}, & \xi < r \leq \zeta \end{cases}$$

$$\text{and } E_t[BackOff] = \sum_{b=1}^r \left( E_t[r \text{ backoff } fs] \cdot P_r^{REQ} failures \right).$$

Finally to compute  $E_t[IDLE]$ , recall that the receivers for potential REQ messages are in the *IDLE* state. Thus, nodes should stay in the *IDLE* state for at least  $E_t[REQ]$ . Moreover, if Jack's REQ towards Chloe collides, she should wait long enough for Jack to

2. We make an implicit assumption that the various packets are of fixed size. If they vary, then an average value will have to be considered.

come back from the *BackOff* to the *REQ* state, at least to try to transmit the request again. Therefore, we consider  $E_t[IDLE]$  to be equal to  $2 \cdot E_t[REQ] + E_t[BackOff]$ , plus a uniformly distributed random interval with values between 0 and  $E_t[REQ]$ . In particular:

$$E_t[IDLE] = 2 \cdot E_t[REQ] + E_t[BackOff] + \frac{E_t[REQ] + 1}{2}.$$

The decision on the duration of this random interval depends entirely on the protocol designer; here we employ the specific strategy for completeness (other strategies could be easily incorporated).

At this stage, we have derived the expected time for being in each state; we may recursively compute  $P_{REQ}$  from Eq. (6), and  $P_{MAC}$  from Eq. (10).

### 3.4.2 Computing $P_{MAI}$

Towards calculating the probability of packet corruption due to multi-access interference, our design computes the probability that a particular pulse is correctly mapped to a ‘0’ or a ‘1’, by comparing the energy levels that appear in the two slots that comprise a  $T_c$  chip. For this, we utilize the Saleh-Valenzuela channel model (discussed in section 1), taking into account all the possible past time instances at which interfering pulses could have been transmitted. The computation of the correct pulse reception, in conjunction with the assumption that majority decoding is used by receivers, enable us derive the probability of bit failure and further the probability of packet corruption due to MAI.

More specifically, let us consider the scenario in Fig. 3. Using TH-BPPM, we assume that the desired signal is transmitted in slot  $S_i$ . Note here that slots  $S_i$  and  $S_{i+1}$  belong to the same chip of interest. In this case, the interference is due to the presence of energy in slot  $S_{i+1}$ . A correlator function at the receiver will compare the energies (plus noise) in the two slots and will decide on whether a ‘0’ or a ‘1’ was transmitted. For mathematical tractability we consider a simplified analysis of a successful reception of a bit, ignoring the receiver functionality and additive Gaussian noise. We assume that the bit is correctly detected if the power in slot  $S_i$  is greater than the power in slot  $S_{i+1}$  by  $g$  dB. As shown in Fig. 3, the desired power (that in slot  $S_i$ ) will be equal to the received power due to the desired pulse plus the sum of powers of all rays appearing within slot  $S_i$ . Moreover, the interference power will be equal to the sum of powers of all rays that arrive during  $S_{i+1}$ .

The  $l_{th}$  cluster starts at  $T_l$  (assuming the first cluster starts at time  $t = 0$  – slot  $S_0$  in our scenario). The  $\nu_{th}$  ray in this cluster is  $\tau_{\nu,l}$  away from  $T_l$ . Given the short range of UWB signals, we assume that pathloss is the same for all the received rays irrespective of the transmitter; the pulse energy  $p^2(t - \tau_{\nu,l})$  is fixed. With this, one can ignore the effects of these factors on the detection probability. In addition, the power (dB) in the  $\nu_{th}$  ray in the  $l_{th}$  cluster is normally distributed with mean  $\mu_{\nu,l}$  (as per Eq. (4)) and standard deviation  $\sigma_{ch}$  (a constant for the channel). The channel coefficients themselves have a lognormal distribution. When we sum the power from different pulses, we have to add lognormal random variables (as described later in this section).

Given a pulse transmission in slot  $S_0$ , there is a cluster that starts in slot  $S_l$  with probability  $P_{cluster}(S_l|S_0)$ . This probability depends on the probability that a previous cluster started in any

of the slots prior to slot  $S_l$ , and is recursively given by:

$$P_{cluster}(S_l|S_0) = \sum_{n=0}^{l-1} \lambda e^{-\lambda(l-n)\frac{T_c}{2}} P_{cluster}(S_n|S_0) \quad (13)$$

Note that  $P_{cluster}(S_0) = 1$ . Given that a cluster starts in slot  $S_l$ , a ray appears in slot  $S_i$  with probability  $P_{ray}(S_i|S_l)$ . In order to find the probability that a ray appears in slot  $S_i$ , we need to find the probability that the previous ray (caused by the same pulse transmission in slot  $S_0$  resulting in a cluster starting in  $S_l$ ) appeared in any slot in the past, between slots  $S_l$  and  $S_{l-1}$ . The first ray (corresponding to the clusterhead) appears in slot  $S_l$ . Hence, the probability that a ray appears in slot  $S_i$  given that the clusterhead occurred in slot  $S_l$  is:

$$P_{ray}(S_i|S_l) = \sum_{j=l}^{i-1} \lambda e^{-\lambda(i-j-1)\frac{T_c}{2}} P_{ray}(S_j|S_l) \quad (14)$$

The power in such a ray is lognormally distributed with the associated normal distribution (in dB) having a mean  $\mu_{i-l,l}$  (as per Eq. (4)) and standard deviation  $\sigma_{ch}$ . Similarly, the probability that a ray occurs in slot  $S_{i+1}$  (i.e.,  $P_{ray}(S_{i+1}|S_l)$ ) can be computed.

The probability that there is a ray in  $S_i$ , but not in  $S_{i+1}$ , given that a cluster started in  $S_l$  is:

$$P_A = (1 - \lambda e^{-\lambda\frac{T_c}{2}}) P_{ray}(S_i|S_l)$$

In this case, the (constructive) interference is lognormal with associated normal parameters  $\mu_{i-l,l}$  and  $\sigma_{ch}$ . The probability that there is no ray in  $S_i$ , but a ray appears in  $S_i + 1$  is:

$$P_B = \sum_{j=l}^{i-1} \lambda e^{-\lambda(i-j)\frac{T_c}{2}} P_{ray}(S_j|S_l) (1 - P_{ray}(S_i|S_l))$$

In this case, the (destructive) interference is again lognormal with associated normal parameters  $\mu_{i-l+1,l}$  and  $\sigma_{ch}$ . Furthermore, the probability that there are rays in both slots  $S_i$  and  $S_{i+1}$  will be:

$$P_C = \lambda e^{-\lambda\frac{T_c}{2}} P_{ray}(S_i|S_l)$$

The interference in this case is the difference of two lognormal random variables  $U$  (with associated normal parameters  $\mu_{i-l,l}$  and  $\sigma_{ch}$ ) and  $V$  (with associated normal parameters  $\mu_{i-l+1,l}$  and  $\sigma_{ch}$ ) i.e., the interference is  $W = U - V$ . We discuss the distribution of  $W$  below. It is also possible that there are no rays in either of the slots  $S_i$  and  $S_{i+1}$ , however that situation does not create any interference.

The distribution of  $W$  can be approximated as lognormal with associated normal parameters  $\mu_W$  and  $\sigma_W$ . There is no closed form solution for the distribution of  $W$ . However, several approximations have been suggested for sums of lognormal random variables. We use the *Wilkinson’s* approach, due to its simplicity [36]. Even though methods with greater accuracy exist, the *Wilkinson’s* approach has been found to provide reasonable cumulative distribution functions for the sum of lognormal random variables [37]. The approach is summarized as follows. Let  $I = \sum_{n=1}^F I_n$  be the sum of  $F$  lognormal random variables  $I_n$ . Then,  $I$  is also lognormal with the associated normal distribution having a mean  $\mu$  and standard deviation  $\sigma$ . The  $\mu$  and  $\sigma$  can be approximated in the following way:

- Let  $X_i = 10 \log_{10} I_i$  be normal with mean  $\mu_{X_i}$  and standard deviation  $\sigma_{X_i}$ . Let  $Y_i = cX_i$  have mean  $\mu_{Y_i}$  and standard deviation  $\sigma_{Y_i}$  where  $c = \ln(10)/10$ .

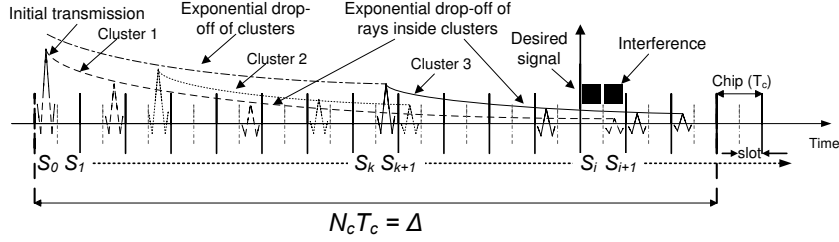


Fig. 3. The multipath delay profile, due to a pulse transmission in slot  $S_0$ . The pulse reception is expected in slot  $S_i$ . Rays appearing in  $S_{i+1}$  may cause interference to the reception of the pulse in the neighboring slot  $S_i$ .

- Let  $X = 10 \log_{10} I$  be the normal distribution associated with the sum  $I$  of the  $F$  lognormal random variables. Then the mean and standard deviation of  $I$  are:

$$\mu_X = \frac{1}{c} (2 \ln u_1 - 0.5 \ln u_2); \quad \sigma_X = \frac{1}{c} \sqrt{\ln u_2 - 2 \ln u_1}$$

where  $u_1 = \sum_{n=1}^F e^{\mu_{Y_n} + \sigma_{Y_n}^2/2}$ , and

$$u_2 = \sum_{n=1}^F e^{2\mu_{Y_n} + 2\sigma_{Y_n}^2} + 2 \sum_{i=1}^{F-1} \sum_{j=i+1}^F e^{\mu_{Y_i} + \mu_{Y_j}} \cdot e^{0.5(\sigma_{Y_i}^2 + \sigma_{Y_j}^2)}$$

assuming that the random variables  $Y_n$  are uncorrelated.

- We denote the Wilkinson's sum of the  $F$  variables as:

$$(\mu_X, \sigma_X) = \bigoplus_{i=1}^F (\mu_{X_i}, \sigma_{X_i})$$

These equations can be modified to account for the difference between two lognormal random variables.

The "average" parameters for the lognormal interference in slot  $S_i$ , conditioned on the clusterhead being in  $S_l$  can now be expressed as

$$\mu_{av|l} = P_A \cdot \mu_{i-l,l} - P_B \cdot \mu_{i-l+1,l} + P_C \cdot \mu_W$$

$$\sigma_{av|l} = P_A \cdot \sigma_{ch} - P_B \cdot \sigma_{ch} + P_C \cdot \sigma_W$$

which is what we use to compute the successful reception of a pulse, described next. Note that these average parameters are a function of the slot  $S_i$ .

As shown in Fig. 4, let  $\Delta = KT_f + LT_c + MT_c$ , so that for any *one* interferer, there are  $K$  or  $K-1$  frames contained completely within  $\Delta$  (depending on the position of slot  $S_i$ ,  $L$  chips of a partial frame before and  $M$  chips of a partial frame after the block of full frames. In addition,  $(L+M)T_c < T_f$ . The frames are numbered 0 through  $K+1$ , and  $S_i, S_{i+1}$  belong to frame 0, as shown in Fig. 4. Consider that an active transmitter has transmissions in each of the full frames and (possibly) the two partial frames. If we assume that a transmission in a chip is equally likely (due to the randomness of the overlapping time hopping sequences), the probability of transmission that could cause interference in slot  $S_i$  in frame 0 is  $M/N_f$  and that in frame  $K+1$  is  $L/N_f$ . In all full frames, this probability is 1.

**Case for full frames:** Consider the transmission of a pulse in some frame  $u$ , where  $u \in [1, K]$ . The pulse transmission occurs in slot  $S_0$  within frame  $u$ , such that the time span between  $S_0$  and  $S_i$  is given by:  $t_u = MT_c + (u-1)T_f + xT_c/2$ , where  $x \in [1, 2N_f]$ . Thus, there are  $\delta = 2M + 2(u-1)N_f + x$  slots between  $S_0$  and  $S_i$  including  $S_0$ . Given that a pulse transmission occurred in  $S_0$ , a clusterhead may appear in any slot  $l \in [0, \delta]$ , with probability

$P_{cluster}(S_l|S_0)$  as per Eq. (13). Since the cluster can begin in any slot between  $S_i$  and  $S_0$ , the average values of the parameters of the lognormal distribution of the interference over all possible clusterhead locations are:

$$\begin{aligned} \mu(S_0) &= \sum_{l=0}^{\delta-1} (P_{cluster}(l|S_0) \mu_{av|l}), \\ \sigma(S_0) &= \sum_{l=0}^{\delta-1} (P_{cluster}(l|S_0) \sigma_{av|l}). \end{aligned} \quad (15)$$

Notice that  $\mu(S_0)$  and  $\sigma(S_0)$  are functions of  $\delta$  and hence, functions of  $x$ . Thus, the average parameters of the lognormally distributed ray power over all possible slots in frame  $u$  are:

$$\mu(u) = \frac{1}{2N_f} \sum_{x=1}^{2N_f} \mu(S_0), \quad \sigma(u) = \frac{1}{2N_f} \sum_{x=1}^{2N_f} \sigma(S_0). \quad (16)$$

There is a similar contribution from transmissions in each of the full frames within  $\Delta$ . The total interference from these transmissions will have a power that is lognormally distributed with average parameters  $\mu_{av\_full}$  and  $\sigma_{av\_full}$ . These are computed using the Wilkinson's approach described earlier:

$$(\mu_{av\_full}, \sigma_{av\_full}) = \bigoplus_{\text{Full Frames}} (\mu(u), \sigma(u)).$$

**Case for partial frames:** We initially assume that the values for  $M$  and  $L$  are known. We consider the case for  $M$  first. The pulse transmission occurs in some slot  $S_0$ , in frame 0, such that the time span between slots  $S_0$  and  $S_i$  is given by:  $t_u = xT_c/2$ , where  $x \in [i, 2M]$ . Thus, there are  $\delta = x$  slots between  $S_0$  and  $S_i$ , including  $S_0$ . Here,  $\mu(S_0)$  and  $\sigma(S_0)$  are computed to be (like in Eq. (16)):

$$\mu_M(u) = \frac{1}{2M} \sum_{x=1}^{2M} \mu(S_0), \quad \text{and} \quad \sigma_M^2(u) = \frac{1}{2M} \sum_{x=1}^{2M} \sigma(S_0)$$

For the case of  $L$ , we have:  $t_u = MT_c + KT_f + xT_c/2$ , where  $x \in [i, 2L]$  and  $\delta = 2M + 2K \cdot N_f + x$ . Moreover:

$$\mu_L(u) = \frac{1}{2L} \sum_{x=1}^{2L} \mu(S_0), \quad \text{and} \quad \sigma_L^2(u) = \frac{1}{2L} \sum_{x=1}^{2L} \sigma(S_0).$$

We set  $\theta = L + M$ . Thus  $\Delta = KT_f + \theta T_c$ . We assume there are  $\varepsilon$  chips between the start of the "0"th frame and  $S_i$ . Depending on  $\varepsilon$ , there are three possible cases.

- Case a:** From Fig. 4 it is clear that  $M = \varepsilon$  and  $L = \theta - \varepsilon$ , where  $\varepsilon \in [1, \theta - 1]$  and there are  $K$  full frames in  $\Delta$ . Considering a fixed value of  $\varepsilon$ , the expected parameters for the lognormally distributed power in slot  $S_i$  can be computed using the Wilkinson's approach:  $(\mu_{av_1}, \sigma_{av_1}) =$

$$= \bigoplus (\mu_{av\_full}, \mu_M(u), \mu_L(u), \sigma_{av\_full}, \sigma_M(u), \sigma_L(u)).$$

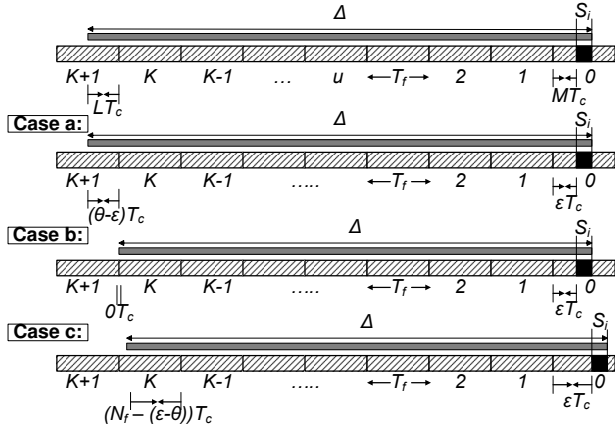


Fig. 4. Handling the cases for which  $\Delta$  covers partially sequence frames.

The above notation indicates that the Wilkinson's approach is used to consider the sum of  $K$  lognormal random variables from the full frames and the two lognormal random variables from the partial frames, to compute the associated normal parameters. These parameters depend upon the value of  $\varepsilon$ . Removing the condition on  $\varepsilon$ :

$$(\mu_{av1}, \sigma_{av1}) = \frac{1}{\theta - 1} \sum_{\varepsilon=1}^{\theta-1} \bigoplus (\mu_{av_{full}}, \mu_M(u), \mu_L(u), \sigma_{av_{full}}, \sigma_M(u), \sigma_L(u)).$$

• **Case b:** As shown in Fig. 4, in this case  $M = \varepsilon = \theta$  and  $L = 0$  (no partial frames before the block of  $K$  full frames). In this case:

$$(\mu_{av2}, \sigma_{av2}) = \frac{\theta}{N_f} \bigoplus (\mu_{av_{full}}, \mu_M(u), \sigma_{av_{full}}, \sigma_M(u))$$

Here, the  $K$  full frames and the partial frame containing  $S_i$  are considered in the Wilkinson's sum.

• **Case c:** In this case the slot  $S_i$  is more than  $\theta$  slots away from the start of frame "0" (i.e.,  $\varepsilon > \theta$ ). We will have  $K - 1$  full frames and two partial frames where  $M = \varepsilon$  and  $L = N_f - (\varepsilon - \theta)$  as shown in Fig. 4. In this case:

$$(\mu_{av3}, \sigma_{av3}) = \frac{1}{N_f - \theta} \sum_{\varepsilon=\theta+1}^{N_f} \bigoplus (\mu_{av_{full}}, \mu_M(u), \mu_L(u), \sigma_{av_{full}}, \sigma_M(u), \sigma_L(u)),$$

where the Wilkinson's sum is computed over  $K - 1$  full frames and the partial frames with  $L$  and  $M$  chips respectively (averaged over the appropriate slots as before). Considering all three cases now, the total average of the normal parameters of the lognormally distributed power at  $S_i$  are:

$$\mu_{avg} = \mu_{av1} \cdot \frac{\theta - 1}{N_f} + \mu_{av2} \cdot \frac{1}{N_f} + \mu_{av3} \cdot \frac{N_f - \theta}{N_f}$$

$$\text{and } \sigma_{avg}^2 = \sigma_{av1}^2 \cdot \frac{\theta - 1}{N_f} + \sigma_{av2}^2 \cdot \frac{1}{N_f} + \sigma_{av3}^2 \cdot \frac{N_f - \theta}{N_f}.$$

Recall that the above calculation involves only one interferer in Chloe's neighborhood. Considering all possible cases of multiple active neighbor transmitters we have:

$$(\mu_{avg_{total}}, \sigma_{avg_{total}}) = \bigoplus_{n=1}^{N-1} (\mu_{avg} \cdot P_{TX}(n), \sigma_{avg}^2 \cdot P_{TX}(n)). \quad (17)$$

Again the Wilkinson's sum of the interfering powers is taken;  $P_{TX}(n)$  (given by Eq. (9)) is the probability that  $n$  transmitters contribute to the interference.

**Impact of Partial Overlaps:** Due to partial overlaps of THSSs, there may be an interfering transmission in the chip of interest. This interfering transmission could be in either slot  $S_i$  or in slot  $S_{i+1}$ , with equal probability. First, the two cases where this interfering transmission occurs in one of the slots and there is no ray in the other slot, tend to compensate for each other, *on average*, in terms of the interference powers. In the last case, the transmission occurs in slot  $S_i$  and a ray due to this transmission appears in slot  $S_{i+1}$ . In this case, due to the proximity of the two signals in time, one may expect the powers to be similar and thus, they together contribute little to the interference power (these effects were implicitly validated via simulations described later).

**Deriving the probability of a pulse failure:** In order for a pulse to be successfully deciphered, the desired power plus the (constructive) interference must be greater than  $g$  dB. This constructive interference is the relative interference power in slot  $S_i$  and was computed as the difference in interference power in slots  $S_i$  and  $S_{i+1}$ . The analysis thus far, assumed that the intended transmission was in slot  $S_i$ ; if on the other hand, the intended transmission was in slot  $S_{i+1}$ , the computed interference acts as *destructive* interference (since it is the relative power in slot  $S_i$ ). Let  $\mu_z$  and  $\sigma_z$  be the mean and standard deviation of the normal distribution associated with the lognormal decision variable  $Z$ , if the intended transmission is in slot  $S_i$ . These values can be obtained by first considering the Wilkinson's sum of the power in the desired pulse ( $P_r$ ) and the lognormal interference. If the intended transmission is in slot  $S_{i+1}$ , the mean and the standard deviation  $\mu'_z$  and  $\sigma'_z$  are computed by considering the difference in the desired pulse ( $P_r$ ) and the lognormal interference computed earlier, using the Wilkinson's approach. Once the corresponding normal parameters in dB are obtained, the value  $g$  can be subtracted from the resulting mean. We are interested in the probability that  $Z > 0$ . Accounting for the fact that the intended transmission could be either in slot  $S_i$  or in slot  $S_{i+1}$  with equal probability, this desired probability is computed as:

$$P_{pulse} = P(Z > 0) = \frac{1}{4} \text{erfc}\left(-\frac{\mu_z}{\sigma_z \sqrt{2}}\right) + \frac{1}{4} \text{erfc}\left(-\frac{\mu'_z}{\sigma'_z \sqrt{2}}\right) \quad (18)$$

The above expression is the probability that one pulse is received successfully in the presence of MAI. Now that we have this expression, we can find the probability that a bit is correctly decoded. Recall that  $PB$  pulses make up a bit (repetition coding). Since the correct decision is obtained by majority decoding, at least half of the  $PB$  pulses have to be correctly received:

$$P(\text{bit success}) = 1 - P(\text{At least } \lfloor \frac{PB}{2} \rfloor \text{ pulses received in error}) =$$

$$= 1 - \sum_{j=0}^{\lfloor PB/2 \rfloor} \left[ \binom{PB}{j} \cdot (P_{pulse}^j) \cdot (1 - P_{pulse})^{PB-j} \right] \quad (19)$$

Finally the probability of packet corruption due to MAI is:

$$P_{MAI} = 1 - [P(\text{bit success})]^{Bits/packet} \quad (20)$$

Having computed both  $P_{MAC}$  and  $P_{MAI}$ , we are able to calculate the probability of packet failure, as per Eq. (5). However, a packet may be lost due to the corruption of either of the REQ, RACK, DATA or ACK transmissions. Let us denote  $P_{MAC}^{<Packet>}$  and  $P_{MAI}^{<Packet>}$  the probability of MAC collision and MAI destruction respectively, for any of the above 4 types of packets. First, the

REQ packet may be lost due to either a MAC collision or a corruption due to MAI. As per Eq. (5), for the REQ packet we have:

$$P_{failure}^{REQ} = P_{MAC}^{REQ} + (1 - P_{MAC}^{REQ}) \cdot P_{MAI}^{REQ}$$

The RACK packet could be lost only due to MAI (i.e., no MAC collision is possible since RACK is transmitted as per a unique THS), and only after the REQ packet is successfully received:

$$P_{failure}^{RACK} = (1 - P_{failure}^{REQ}) \cdot P_{MAI}^{RACK}$$

Furthermore, the DATA packet could also be lost only due to MAI, and only after REQ and RACK are successfully received:

$$P_{failure}^{DATA} = (1 - P_{failure}^{REQ}) \cdot (1 - P_{failure}^{RACK}) \cdot P_{MAI}^{DATA}$$

Similarly, the ACK could be lost only due to MAI:

$$P_{failure}^{ACK} = (1 - P_{failure}^{REQ}) \cdot (1 - P_{failure}^{RACK}) \cdot (1 - P_{failure}^{DATA}) \cdot P_{MAI}^{ACK}$$

Finally, the transmission process for a DATA packet as per the considered MAC could fail if just one of the above packets fails. Hence:

$$P_{failure} = P_{failure}^{REQ} + P_{failure}^{RACK} + P_{failure}^{DATA} + P_{failure}^{ACK}. \quad (21)$$

Next, we compute the per node saturation throughput. This throughput is defined to be the product of (i) the fraction of time for which a node is able to send useful pulses and (ii) the raw data rate; the throughput is expressed in units of bits/sec. The duration during which useful data is transferred is:

$$\frac{(1 - P_{failure}(DATA)) \cdot E_t[DATA] \cdot DATA}{DATA + RACK + ACK + BEACON}.$$

However, the transmitter is able to send a pulse only once every sequence frame; therefore, this time has to be scaled down by a factor  $N_f$ . If the raw data rate of the channel is  $Rate$ , the achievable rate is scaled down by a factor  $PB$  due to the use of repetition codes. Thus, using Eq. (11):

$$T_{put} = \frac{DATA \cdot (1 - P_{failure}) \cdot Rate}{E_t[REQ] + E_t[DATA] + E_t[BackOff] + E_t[IDLE]}. \quad (22)$$

## 4 EVALUATING OUR FRAMEWORK

In this section, we evaluate the accuracy of CTU and discuss how CTU can guide MAC protocol design decisions.

### 4.1 Validating the accuracy of CTU

We compare the analytical results produced by CTU with the simulated behavior of the network. The considered input parameters are summarized in Table 1. In more detail: (1) We perform an extensive set of simulations of the network, using a C++ simulation platform [1]. We have incorporated the lognormal fading distribution in the SV model for the various pulse transmissions, as per [5]. Our modifications to the simulator consist of: (a) generating the Poisson-based arrival of clusters and rays, for a period  $\Delta$  ( $N_c$  chips) after each pulse transmission (Eq. (1), (2)), and (b) the impulse response of transmitted pulses (as per Eq. (3)). (2) We implement CTU in both C++ and Matlab, and we use it to compute the probability of packet transmission failure. We then estimate the average MAC layer throughput per node, as per Eq. (22). The Stirling's approximation [38] was used to compute factorials and this considerably speeds up the computations. The set of all our analytical results are obtained in less than one hour,

Parameter	Value
Tc	1 nsec
A	80×80 m <sup>2</sup>
R	11 m
N <sub>f</sub>	50
PB	120
ξ	6
J	256
Bandwidth	7.5 GHz
DATA	2048 bits
ACK	120 bits
REQ	120 bits
RACK	120 bits

Parameter	Value
Γ	14.93
γ	7.03
Λ	0.0667
λ	3
P <sub>m</sub>	5.59 nW
ζ	15
Rate	110 Mbps
σ	4.8 dB
g	10 dB
N <sub>c</sub>	120
BEACON	32 bits
Sim. duration	10 <sup>8</sup> slots

TABLE 1  
Simulation parameters, when not variable

while the simulations take many days. Note that throughout our numerical experiments, we observed that convergence with CTU is fast: the iterative operations invoked by our framework (see section 3.1) typically yield converged results in less than 200 rounds.

We compare the results produced by CTU and the simulator, in terms of per node throughput, on average, for different values of  $N_f$ . Figs. 5 and 6 depict this comparison, for the cases with  $N_f =$  and , respectively, and for  $PB = 120$ . We observe that the results from CTU are very close to those from the simulator; the throughputs corresponding to CTU are shifted only by a few nodes with regards to the throughputs computed by the simulator, as seen in Fig. 7<sup>3</sup>. The discrepancy in the results are due to: (a) the approximations from the Wilkinson's approach and (b) while a constant pathloss of 20 dB is assumed in the analysis, the Friis path loss model [39] is used in the simulations. Next, we use CTU to examine the impact of system and environmental factors on the performance.

### 4.2 On the impact of system parameters on MAI

We first examine the impact of system parameters such as  $N$  (the number of nodes in the network),  $N_f$  and  $PB$  on the MAI. Fig. 8 depicts the probability of a pulse success versus  $N$ . Interestingly, as  $N$  initially increases, the probability of pulse success also increases. This contradicts the common intuition that fewer pulses are likely to be successful as  $N$  increases. This is a direct consequence of the impact of the number of interfering users in the receiver's (Chloe's) vicinity and in particular Eq. (17). The MAI, as seen from Eq. (17), is dependent on the number of active interfering transmitters. When  $N$  is small, the number of nodes in Chloe's range is even smaller. As the number of neighbors increases initially, the transmitters will compete among each other for the few available receivers and will often end up in the *BackOff* state. Thus, the number of active transmitters actually reduces and as a result, the MAI reduces as well. Note that this happens with few nodes. However, beyond a certain density, the nodes are able to find receivers and thus, are able to stay as "active" transmitters. From this point on, the MAI begins to increase again. The point at which the transition occurs depends on the area of deployment ( $A$ ), the transmission range, and  $N$ . As one might expect, this point is somewhat independent of the value of  $N_f$ ; the decrease is due to the inability of transmitters to find receivers and not because of collisions due to overlaps between time-hopping

3. In Fig. 7, the supported number of users is the node population for which the average per node throughput is the maximum. Adding more nodes in the network will further decrease the node throughput due to contention and interference.

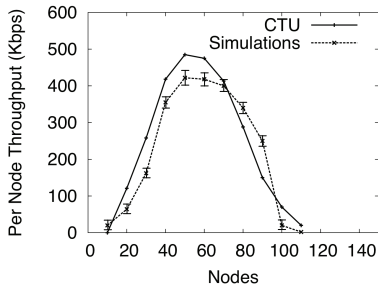


Fig. 5. Average throughput per node, for  $N_f = 10$  and  $PB = 120$ .

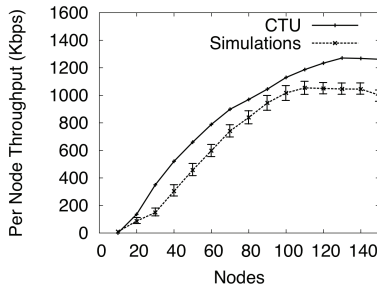


Fig. 6. Average throughput per node, for  $N_f = 20$  and  $PB = 120$ .

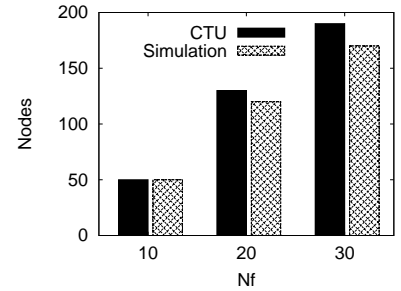


Fig. 7. CTU accurately estimates the supported number of users.

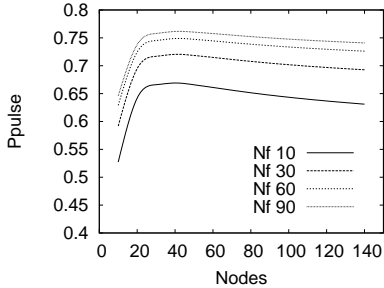


Fig. 8. The probability of pulse success increases with  $N_f$ .

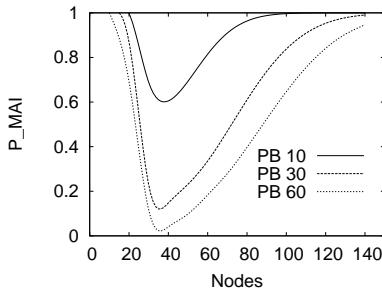


Fig. 9. Larger values of  $PB$  result in reduced interference.

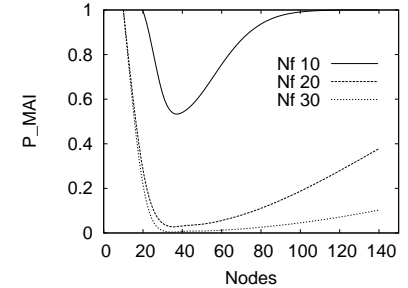


Fig. 10. The value of probability  $P_{MAI}$  reduces with  $N_f$ .

sequences. Note here that this results due to the MAC protocol design choice of selecting receivers at random. It is possible that a cleverer receiver selection strategy, adopted by another potential protocol, will not present such a behavior.

Notice that the value of  $N_f$  clearly determines the extent of interference. The larger the value of  $N_f$ , the smaller is the amount of destructive interference experienced by receivers. In addition to  $N_f$ , the chosen  $PR$  value impacts the achieved performance. In Fig. 9, we plot the reduction in  $P_{MAI}$ , as the value of  $PR$  increases. We observe that by increasing the redundancy in the number of transmitted pulses per bit, the negative impact of interference is decreased and hence the probability of packet failure due to interference is reduced. Extremely high  $N_f$  values impact the network throughput, due to significant delays between pulse transmissions [30]. Finally,  $P_{MAI}$  varies conversely with  $P_{pulse}$ , as depicted in Fig. 10.

### 4.3 On the behavior of the MAC layer

Next, we examine the MAC layer success probability  $P_{MAC}$  with the simple MAC procedure considered (Fig. 11). With small  $N$ , transmitters have a very limited set of potential receivers to choose from. This causes a significant number of REQ collisions; therefore  $P_{MAC}$  initially has a very high value. As  $N$  increases, more and more receivers become available; this reduces the probability that a receiver is selected by more than one transmitter. The level of interference affects the value of  $P_{MAC}$  (since REQ messages may suffer interference as seen in Eq. (12)). From Figs. 10 and 11, we observe that as  $N$  increases,  $P_{MAI}$  increases as well, and drives  $P_{MAC}$  higher. Thus,  $P_{MAC}$  does not drop sharply (but gradually) with the increased  $N$ .

Since the control messages (REQ, RACK, ACK, BEACON) are of short length, compared to DATA packets, the impact of MAI on them is slightly less severe. This is shown in Fig. 12. The probability of REQ transmission failure is iteratively computed

by Eq. (5) and (12) and hence is dependent on  $P_{MAI}$  for the REQ message. Notice that for the different values of  $N_f$ , the values of  $P_{MAC}$  and  $P_{failure}(REQ)$  change.

The combined behavior of  $P_{MAC}$  and  $P_{MAI}$  determines the average achievable MAC layer throughput, as per Eq. (22). As shown in Figs. 5 and 6, for very small  $N$ , the average per node throughput is very low, since  $P_{MAC}$  and  $P_{MAI}$  are high. For moderate populations (40 to 70 nodes) the average throughput is high; however as  $N$  increases further, the throughput decreases due to increased  $P_{MAI}$ .

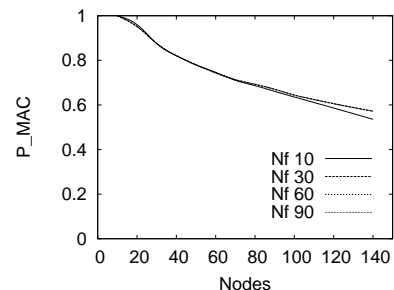


Fig. 11. The value of  $P_{MAC}$  is not significantly affected by the length of  $N_f$ , for different number of nodes.

## 5 USING CTU TO COMPARE DIFFERENT MAC PROTOCOLS

The modular nature of our framework allows one to tune it to evaluate a wide class of different TH-BPPM based MAC protocols as well as different PHY considerations. Next, in subsections 5.1 to 5.3, we demonstrate how CTU can be adjusted to facilitate the performance evaluation of different MAC protocols, for impulse-radio UWB. In particular, we describe a different MAC protocol (which we call **M2**), we implement it in CTU and we conduct numerical experiments to observe the performance of this new

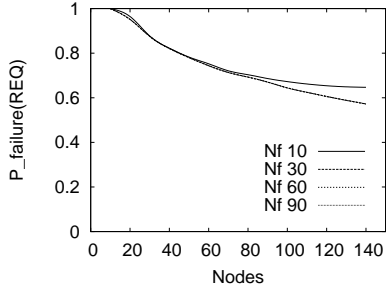


Fig. 12. The probability of REQ packet failure depends on the MAC operations.

protocol; we discuss the CTU results and compare this MAC protocol with the one described in section 3. Similarly as before, we assume that the traffic is fully-saturated.

### 5.1 A quick description of the MAC protocol

We consider a different MAC procedure that involves only DATA and BEACON packet transmissions. The new state diagram for this protocol is shown in Fig. 13. With this MAC, Jack uses

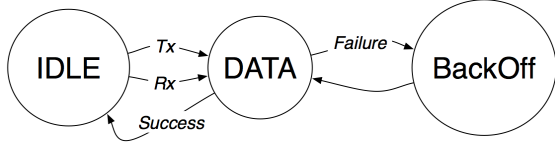


Fig. 13. The state diagram for the M2 MAC protocol. The DATA packet is transmitted as per the receiver's THS.

Chloe's THS to transmit his DATA packet, without sending a REQ message first; Chloe will realize that Jack wishes to send a packet to her by decoding an appropriate preamble; she tries to receive the data packet without sending a RACK message to Jack. As shown in Fig. 13, as soon as Chloe starts receiving the DATA packet, she transits from the *IDLE* to the *DATA* state along with Jack. If the DATA transmission is successful, Chloe will transmit her short BEACON frame, thereby informing her neighbors that she is now idle. By receiving Chloe's BEACON message, Jack will know that his DATA packet was received intact; he will further transmit his BEACON frame, as well. On the other hand, if the DATA transmission is corrupted (either due to a MAC collision or due to MAI), Jack will go to the *BackOff* state, and he will retry his transmission after his backoff counter is zero, as per the process described in section 3.

### 5.2 Adjusting CTU to facilitate the new MAC scheme

In order to apply the CTU framework to the afore-described protocol, the following six adjustments would have to be performed:

**I)** With this MAC protocol there is no *REQ* state, therefore  $P_{REQ} = 0$  in Eq. (6), since  $E_t[REQ] = 0$  in Eq. (12). Moreover, no ACK or RACK messages are involved, and thus their contribution is ignored in Eq. (6) and (11); in addition, the probability of packet failure does not consider the successful transmissions of those frames. Moreover, BEACON frames again do not collide.

**II)** Since DATA is transmitted as per the receiver's THS and since there are no REQ/RACK transmissions, the DATA packet is not protected from MAC level collisions. In other words, since

Jack is using Chloe's THS for the DATA packet, it is possible that his DATA transmission collides with other possible DATA transmissions towards Chloe. Thus, it may take a number of DATA transmission failures until the DATA packet is successfully delivered (if delivered at all). Clearly the inter-state transition will still be a Markov process, hence we will have:

$$P_{DATA} = \frac{E_t[DATA]}{E_t[DATA] + E_t[BackOff] + E_t[IDLE]},$$

where  $P_{DATA}$  is the probability of being in the *DATA* state (including successful and retransmission attempts). Meanwhile the probability  $P_{Chloe}$  that no other transmitters (besides Jack) pick Chloe will now depend on  $P_{DATA}$  instead of  $P_{REQ}$ . Moreover,  $P_{tx} = P_{DATA}$  in Eq. (9).

**III)** Upon failure of the DATA packet transmission, Jack will go to the *BackOff* state and he will retry the transmission as soon as his back-off timer expires. The duration of each retransmission is  $t_r = N_f \cdot PB \cdot DATA$ ; and for the successful transmission it is  $t_s = N_f \cdot PB \cdot (DATA + BEACON)$ . The expected time in the *DATA* state considering  $r$  transmission failures for the DATA packet will be:

$$E_t[DATA] = \sum_{r=0}^{\zeta} \left[ (r \cdot t_r + t_s) \cdot P_r^{DATA} \cdot (1 - P_{failure}^{DATA}) \right]. \quad (23)$$

**IV)** The expected time in the *BackOff* state for this MAC protocol now depends on the probability of transmission failure of the DATA packet,  $P_{failure}^{DATA}$ . Thus, the expected time in the *BackOff* state will be:

$$E_t[BackOff] = \sum_{b=1}^r \left( E_t[r \text{ backoffs}] \cdot P_{failure}^{DATA} \right). \quad (24)$$

Note that the time  $E_t[r \text{ backoffs}]$  is computed as in section 3.

**V)** Regarding  $E_t[IDLE]$ , we want the receivers to wait for some time for transmitters to return from the *BackOff* state, to try the DATA transmission again. Recall that as soon as a packet transmission is initiated with M2, both transmitter and receiver transit to the *DATA* state; in contrast with M1, while the transmitter is in the *REQ* state, the receiver is still in the *IDLE* state. Thus for M2 we consider that the receiver will wait in the *IDLE* state for the duration of a BackOff period, i.e.:

$$E_t[IDLE] = E_t[BackOff].$$

**VI)** The probability of packet failure involves DATA packets only. As per Eq. (21), it will now be:

$$P_{failure} = P_{failure}^{DATA} = P_{MAC}^{DATA} + (1 - P_{MAC}^{DATA}) \cdot P_{MAI}^{DATA}. \quad (25)$$

Note again that for Eq. (23) and (24),  $P_{failure}^{DATA}$  will be computed recursively through Eq. (25).

From this analysis it becomes clear that CTU can be easily tweaked to facilitate the evaluation of different MAC protocols for impulse-radio UWB networks.

### 5.3 Assessing the behavior of M2 using CTU

We embed the M2 protocol in our framework implementation and we perform numerical experiments to assess its behavior. In what follows, we provide a comparison between the behavior of M2 and the one of the MAC procedure **M1**, described in section 3.

First we compare the probability of failure of the DATA packet transmission with the two protocols. In particular, we are

interested in the percentage increase in  $P_{failure}$ , when replacing M1 with M2. This increase is depicted in Fig. 14 for the case of 50 nodes and  $N_f = 50$ . We observe that the difference in the probability of failure increases, as we increase the value of  $PB$ . With M2 the DATA packet is constantly more prone to failure. This is because the packet is not protected from MAC collisions, since REQ and RACK packets are absent. Increasing the value of  $PB$  decreases the probability of failure for both protocols, since  $P_{MAI}$  decreases. However, this favors the performance of M1 more than M2. The absence of the protective REQ and RACK messages seems to be tremendously degrading the performance of M2, which cannot be substantially improved through the increment of the value of  $PB$ . In contrast, M1 is already more robust to prolonged DATA failures, due to the REQ/RACK mechanism, and hence there is more space for performance improvement by the increment of  $PB$ .

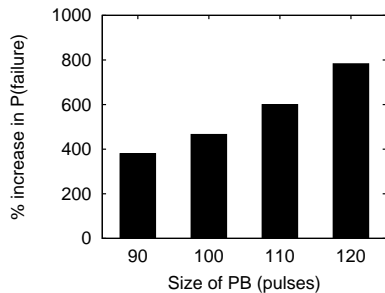


Fig. 14. Substituting M1 with M2 incurs a tremendous increase in  $P_{failure}$ , which increases for increasing  $PB$  values.

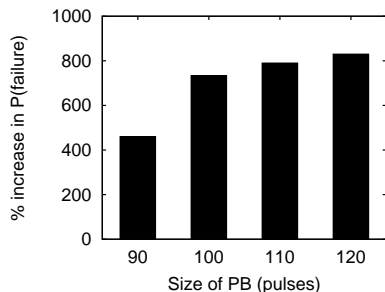


Fig. 15. Although increasing  $N_f$  reduces failures, the replacement of M1 with M2 increases  $P_{failure}$ .

Moreover, as in section 4, we vary the length of the sequence frame  $N_f$  in order to observe how  $P_{failure}$  varies. Fig. 15 depicts the results for the case of 50 users and  $PB = 120$ . Similarly as above, we observe that even though higher values of  $N_f$  reduce the probability of failure for both protocols, M1 always manages to protect the DATA packet transmission better than M2.

Hence, we have demonstrated that CTU can easily accommodate a plurality of variations of our basic MAC procedure. The incorporation of a different MAC protocol requires simple modifications mainly in the part that computes  $P_{MAC}$ ; i.e., for the given PHY implementation, no changes are required in the computation of  $P_{MAI}$ .

## 6 SCOPE AND LIMITATIONS OF CTU

In this section, we elaborate on particular assumptions made during the design of CTU, we discuss further potential adjustments and we deliberate on the limitations of our framework.

### 6.1 On the assumption of constant path-loss

As described in [2], there are two approaches for looking at UWB channel measurements. The first approach just approximates “typical” locations of UWB devices (in effect ignoring path loss and concentrating on multipath only) and the second approach tries to provide a comprehensive statistical model (for example using different path loss exponents for LOS and NLOS conditions, sometimes for different parts of the UWB band). In the 802.15.3 model, two different path-losses are used – 20 dB at a receiver distance of 10m from the transmitter and 12 dB at a receiver distance of 4m from the transmitter [5]. Further, as noted in [2], in the measurement process, it is difficult to separate the large-scale and small-scale fading characteristics. Thus the log-normal fading in the S-V models derived from measurements and used in the paper implicitly include the large-scale fading effects. In our simulations, we used a fixed path-loss of 20 dB for all pairs of transmitters and receivers to reduce the complexity (e.g., the model will have to consider the exact distances and also the nature of the channel – LOS or NLOS between each transceiver pair). Another way of interpreting the fixed path-loss would be that some interferers will have a larger path-loss, while others will have a smaller path-loss and using the same path-loss for all transceiver pairs averages out this effect. A more accurate pathloss model can also be integrated when computing the probability of successful reception of a pulse in Eq. 18; however, one would have to compute the average interference, conditional on the channel attenuations to the different users.

### 6.2 Other possible adjustments

As we showed earlier, various parameters such as the expected times in each state can be tuned as appropriate to the specific MAC protocol under consideration, in order to derive  $P_{MAC}$  as per Eq. (10). This is easier when the MAC operations are captured through a finite state machine. More than that, consider a case where the packet sizes vary and the back-off algorithm is different from the one considered. These features can be easily incorporated into CTU by appropriately modifying the specific portions of the framework. As further examples:

- 1) While developing CTU, we assumed that nodes have a radial range  $R$ . However, note that this assumption is primarily used to (a) fix a pathloss value and (b) determine the number of possible interferers. One can easily incorporate a different range distribution in the CTU framework. The main change required will be in Eq. (7).
- 2) One may incorporate a different (than the SV) channel model, as long as the model describes the ray inter-arrival times and gains. For example, adopting the Nakagami fading distribution instead of the lognormal distribution [5] is possible. In addition, one may consider a model where clusters are fading independently or rays. In such a case, Eq. (4) could be easily modified to include a fading term with regards to the cluster arrival and a fading term for the ray arrival [5].
- 3) We assume PPM in this paper. If BPSK is used, the polarity of the pulse will be different depending on whether a ‘0’ or ‘1’ is to be sent. In such a case, instead of looking at the energy difference between the time slots (as in PPM), one has to simply look at whether the energy received in a

slot is positive or negative. With minor modifications, our analysis can be applied with BPSK.

### 6.3 Limitations of CTU

The CTU framework however, has the following limitations. **(a)** CTU does not account for the use of convolutional or block codes. Incorporating the effects of coding will increase the complexity of CTU. **(b)** We have based our analytical framework on TH-BPPM based communications. Thus, other multiple access strategies (such as M-PPM, CDMA or OFDMA) are not applicable. As an alternative transmission technique, the Wimedia Alliance supports an OFDM (Orthogonal Frequency Division Multiplexing) based specification [40] for UWB.

## 7 CONCLUSION

We design and evaluate CTU, an analytical framework for MAC layer throughput estimation in impulse-based multihop UWB networks. The modularity of CTU enables one to easily tune it to model a wide range of MAC protocols; the only requirements are that the underlying modulation scheme be BPPM and time-hopping be used as the multi-access procedure. CTU is the first analytical framework that captures the cross layer interactions between the PHY and the MAC layers. We validate the accuracy of CTU with extensive simulations and we observe interesting properties through a plurality of numerical results. Finally, we use CTU to analyze a different MAC protocol, which we compare against the default protocol that we use. The analysis suggests that CTU can be easily modified to host the design of different TH-BPPM MAC protocols.

**Acknowledgment:** We wish to thank the Associate Editor, Professor Sunghyun Choi, and the anonymous reviewers for their constructive comments.

## REFERENCES

- [1] I. Broustis, S. Krishnamurthy, M. Faloutsos, M. Molle, and J. Foerster. Multiband Media Access Control in Impulse-Based UWB Ad hoc Networks. In *IEEE Trans. on Mobile Comp.*, Vol. 6, No. 4, Apr. 2007.
- [2] A. Molisch, J. R. Foerster, and M. Pendergrass. Channel Models for Ultra-wideband Personal Area Networks. In *IEEE Wireless Comm. Mag.*, Vol. 10, No. 6, Dec. 2003.
- [3] M. Z. Win and R. A. Scholtz. Impulse Radio: How It Works. In *IEEE Comm. Letters*, Vol.2, No.1, Jan. 1998.
- [4] A. Saleh and R. Valenzuela. A statistical model for indoor multipath propagation. In *IEEE JSAC*, Vol. SAC-5, No. 2, Feb. 1987.
- [5] J. R. Foerster. Path Loss Proposed Text and S-V Model Information. In *Intel R&D document*, IEEE P802.15-02/xxxr0-SG3a, Sept. 2002.
- [6] J. Y. Le Boudec, R. Merz, B. Radunovic, and J. Widmer. DCC-MAC: A Decentralized MAC Protocol for 802.15.4a-like UWB Mobile Ad-Hoc Networks Based on Dynamic Channel Coding. In *Broadnets*, 2004.
- [7] J. G. Proakis. Digital Communications, 4th Edition. McGraw-Hill, 2001.
- [8] H. Hashemi. The Indoor Propagation Channel. Proceedings of the IEEE, Vol. 81, No. 7, pp. 943-968, 1993.
- [9] S. Verdú. Maximum Likelihood Sequence Detection for Intersymbol Interference Channels: A New Upper Bound on Error Probability. *IEEE Trans. on Information Theory*, Vol. 33, No. 1, Jan. 1987.
- [10] G. S. Muller and C. K. Pauw. Trellis Code Modulation on a Rayleigh Fading Channel. In *IEEE ICC*, 1989.
- [11] M. V. Eyuboglu and S. U. H. Qureshi. Reduced state sequence estimation for coded modulation on intersymbol interference channels. *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 6, pp. 989995, 1989.
- [12] H. Vikalo and B. Hassibi. Maximum-Likelihood Sequence Detection of Multiple Antenna Systems over Dispersive Channels via Sphere Decoding. *EURASIP Journal on Applied Signal Processing* 2002:5, 525531.
- [13] R. Merz, J.-Y. Le Boudec, and S. Vijayakumaran. Effect on Network Performance of Common versus Private Acquisition Sequences for Impulse Radio UWB Networks. In *IEEE ICUBW*, 2006.

- [14] R. Merz and J.-Y. Le Boudec. Performance Evaluation of Impulse Radio UWB Networks using Common or Private Acquisition Preambles. In *IEEE Trans. on Mobile Comp.*, Vol. 8, Iss. 3, 2009.
- [15] G. Giancola and M.-G. Di Benedetto. A Novel Approach for Estimating Multi-User Interference in Impulse Radio UWB Networks: The Pulse Collision Model. In *Elsevier Signal Processing (special issue on Signal Processing in UWB communications)*, Vol. 86, pp. 2185-2197, 2006.
- [16] I. Guvenc, H. Arslan, S. Gezici, and H. Kobayashi. Adaptation of Multiple Access Parameters in Time Hopping UWB Cluster Based Wireless Sensor Networks. In *IEEE Mobile Adhoc Sensor Systems Conf. (MASS)*, Fort Lauderdale, FL, pp. 235-244, 2004.
- [17] I. Broustis, M. Molle, S. Krishnamurthy, M. Faloutsos, and J. Foerster. A New Binary Conflict Resolution Based MAC protocol for Impulse-Based UWB Ad hoc Networks. In *Wireless Comm. and Mobile Comp.*, Vol. 6, Issue 7, Nov. 2006.
- [18] R. Merz, J. Y. Le Boudec, J. Widmer, and B. Radunovic. A Rate-Adaptive MAC Protocol for Low-Power Ultra-Wide Band Ad hoc Networks. In *Ad Hoc Now*, 2004.
- [19] R. Merz, J. Widmer, J. Le Boudec, and B. Radunovic. A Joint PHY/MAC Architecture for Low-Radiated Power TH-UWB Wireless Ad Hoc Networks. In *Wireless Comm. and Mobile Computing*, Vol. 5, No. 5, August 2005.
- [20] R. Jurdak, P. Baldi, and C. V. Lopez. U-MAC: a Proactive and Adaptive UWB Medium Access Control Protocol. In *Wireless Comm. and Mobile Computing*, Vol. 5, No. 5, pp. 551-566, August 2005.
- [21] F. Cuomo, C. Martello, A. Baiocchi, and C. Fabrizio. Radio Resource Sharing for Ad Hoc Networking with UWB. In *IEEE JSAC*, December 2002.
- [22] M.-G. Di Benedetto, L. De Nardis, M. Junk, and G. Giancola. (UWB)<sup>2</sup>: Uncoordinated, Wireless, Baseborn Medium Access for UWB Communication Networks. In *Mobile Networks and Applications 10*, 663674, 2005.
- [23] J. Ding, L. Zhao, S. Medidi, and K. M. Sivalingam. MAC Protocols for Ultra-Wide-Band Wireless Networks: Impact of Channel Acquisition Time. In *SPIE-ITCOM*, December 2002.
- [24] A. Hicham, Y. Souilmi, and C. Bonnet. Self-balanced receiver-oriented MAC for ultra-wide band mobile ad hoc networks. In *Ultra Wideband Systems*, June 2003.
- [25] S. Kolenchery, J. Townsend, and J. Freebersyser. A Novel Impulse Radio Network for Tactical Military Wireless Communications. In *IEEE MILCOM*, 1998.
- [26] B. Radunovic and J. Y. Le Boudec. Optimal Power Control, Scheduling and Routing in UWB Networks. In *IEEE JSAC*, Vol. 22, No. 7, September 2004.
- [27] N. J. Agusut and D. S. Ha. An Efficient UWB Radio Architecture for Busy Signal MAC protocols. In *IEEE SECON*, 2004.
- [28] R. Merz and J. Y. Le Boudec. Conditional Bit Error Rate for an Impulse Radio UWB Channel with Interfering Users. In *IEEE ICU*, 2005.
- [29] J. A. Gubner and K. Hao. A Computable Formula for the Average Bit Error Probability as a Function of Window Size for the IEEE 802.13a UWB Channel Model. In *IEEE Trans. Micr. Theory*, Vol. 54, No. 4, 2006.
- [30] I. Broustis, A. Vlavianos, and S. V. Krishnamurthy. On the MAC Layer Performance of Time-Hopped UWB Ad Hoc Networks. In *ICCCN*, 2006.
- [31] D. Cassioli, M. Z. Win, and A. Molisch. The Ultra-wide Bandwidth Indoor Channel: From Statistical Model to Simulations. In *IEEE JSAC*, Vol. 20, No. 6, Aug. 2002.
- [32] S. S. Ghassemzadeh et al. UWB Indoor Delay Profile Model for Residential and Commercial Environments. In *IEEE VTC*, 2003.
- [33] L. J. Greenstein et al. Comparison Study of UWB Indoor Channel Models. In *IEEE Trans. of Wireless Comm.*, Vol. 6, Issue 1, Jan. 2007.
- [34] A. Gupta and P. Mohapatra. A Survey on Ultra Wide Band Medium Access Control Schemes. In *Tech. Report CSE-2006-4*, UC Davis, 2006.
- [35] S. M. Ross. Applied Probability Models With Optimization Applications. Dover Publications, 1992.
- [36] P. Cardieri and T. S. Rappaport. Statistics of the Sum of Lognormal Variables in Wireless Communications. In *IEEE VTC*, 2000.
- [37] N. Beaulieu, A. A. Abu-Dayya, and P. J. McLane. Estimating the Distribution of a Sum of Independent Lognormal Random Variables. In *IEEE Trans. Comm.*, Vol. 43, No. 12, Dec. 1995.
- [38] M. Abramowitz and I. Stegun. Handbook of Mathematical Functions. In <http://www.math.sfu.ca/~cbm/aands>, 2006.
- [39] S. Roy, J. R. Foerster, V. S. Somayazulu, and D. G. Leeper. Ultrawideband Radio Design: The Promise of High-Speed, Short-Range Wireless Connectivity. In *Proc. of the IEEE*, Feb. 2004.
- [40] WiMedia Alliance. <http://www.wimedia.org>.



**Ioannis Broustis** Ioannis Broustis is currently with Alcatel-Lucent, NJ. He received his M.Sc. and Ph.D. degrees in Computer Science and Engineering from the University of California, Riverside, and his Diploma in Electronics and Computer Engineering from the Technical University of Crete, Greece. During 2008, he was a Researcher with the Center for Research and Technology Hellas, Thessaloniki, Greece, as well as an Adjunct Assistant Professor with the University of Thessaly, Volos, Greece. During 2006, he was with Intel Research in Cambridge,

U.K., while during 2005, he was with Nokia R&D, Boston, MA. His research interests include LTE security, cross-layer network protocol design, development, and testbed experimentation for wireless networks.



**Angelos Vlavianos** Angelos Vlavianos received his diploma from the Department of Electronics and Computer Engineering at the Technical University of Crete, Greece and his M.Sc from the Department of Computer Science and Engineering, at the University of California Riverside. His interests include Internet Video Steaming, assessing performance in wireless networks, peer-to-peer networks and ad-hoc networks. Currently Angelos Vlavianos is the co-founder of 7Layers Co. in Greece, which specializes in Internet Video Streaming and tailor-made

applications for Enterprises.



**Prashant Krishnamurthy** Prashant Krishnamurthy is an associate professor in the School of Information Sciences, University of Pittsburgh (Pitt). At Pitt, he regularly teaches courses on wireless networks, cryptography, and network security. His research interests are in wireless network security, wireless data networks, position location in indoor wireless networks, and radio channel modeling for indoor wireless networks. He also served as the chair of the IEEE Communications Society Pittsburgh Chapter from 2000 to 2005. He is a member of the IEEE.



**Srikanth V. Krishnamurthy** Srikanth V. Krishnamurthy received his Ph.D degree in electrical and computer engineering from the University of California at San Diego in 1997. From 1998 to 2000, he was a Research Staff Scientist at the Information Sciences Laboratory, HRL Laboratories, LLC, Malibu, CA. Currently, he is a Professor of Computer Science at the University of California, Riverside. His research interests are primarily in wireless networks, network security and Internet technologies.

Dr. Krishnamurthy is the recipient of the NSF CAREER Award from ANI in 2003. He has also co-edited the book "Ad Hoc Networks: Technologies and Protocols" published by Springer Verlag in 2005. He was the editor-in-chief for ACM MC2R from 2008-2009 and is a senior member of the IEEE.