Concurrent Transmissions for Multi-hop Communication on Ultra-wideband Radios

Diego Lobba, Matteo Trobinger, Davide Vecchia, Timofei Istomin, Gian Pietro Picco

University of Trento, Italy

{diego.lobba, matteo.trobinger, davide.vecchia, timofei.istomin, gianpietro.picco}@unitn.it

Abstract

Concurrent transmissions, as popularized by Glossy, have proven an effective, state-of-the-art technique for the design of reliable and efficient network protocols. However, their exploitation is largely confined to IEEE 802.15.4 narrowband radios. In this paper, we investigate the extent to which concurrent transmissions can be applied to ultra-wideband (UWB) radios, whose popularity is rapidly growing.

We adopt a system-driven approach, where techniques and codebases representative of the state of the art are adapted for UWB and evaluated in a 23-node indoor testbed yielding multi-hop topologies. We show that, once embodied in a full-fledged system, UWB concurrent transmissions yield benefits similar to narrowband, i.e., near-perfect reliability and very low latency and energy consumption, along with order-of-magnitude improvements in network-wide time synchronization. Further, our implementations suggest that existing higher-level protocols built atop Glossy require only minimal adaptation.

Our results pave the way for the exploitation of concurrent transmissions in UWB, which we foster by releasing our systems as open source, enabling their immediate use and improvement by researchers and practitioners.

1 Introduction

Ultra-wideband (UWB) radios are rapidly becoming a prominent player in the ever-changing landscape of Internet of Things (IoT) enabling technologies. Their reliance on very short impulses yields *i)* distance estimation (ranging) with significantly higher accuracy (centimeters vs. meters) than competing RF-based technologies, and *ii)* high-rate wireless communication, therefore reuniting in a single radio transceiver two key functions of many IoT scenarios.

Motivation. Nevertheless, a staple network stack for UWB is still missing. This is partly explained by the fact that the interest in UWB, at its peak about a decade ago and largely forgotten thereafter, renewed only recently, fueled by new chips (e.g., the popular DecaWave DW1000 [6]) that yield high ranging accuracy and yet are small, cheap, energy-savvy, and standard-compliant. In contrast, during the same decade, research in academia and industry generated numerous protocols, systems, and real-world deployments targeting a variety of traffic patterns, operating conditions, and stack layers. Among these, the approaches based on *concurrent transmissions* on the same radio channel, as popularized by Glossy [8], have proven a very effective building block for protocol design. Several protocols (e.g., [8, 7, 10, 13, 15]) embraced this technique and its variants, ultimately pushing the envelope of IEEE 802.15.4 radios by achieving low latency, high reliability, low energy consumption—all at once.

Besides published works, the fact that almost all of the teams (and *all* of the top ones) in the four editions of the EWSN Dependability Competition relied on Glossy-like systems is another witness that concurrent transmissions are the state-of-the-art technique in IEEE 802.15.4 narrowband radios. Therefore, it is natural to investigate whether they are applicable also to UWB radios. We provide a concise primer of both UWB and concurrent transmissions in §2.

Goals. As we discuss in the context of related work (§3), a recent study [19] elicited the conditions for successful UWB concurrent transmissions, both on the same channel (as in our case) and on different channels (not of interest here). Moreover, in the UWB localization system in [12], the use of Glossy-like concurrent transmissions is reported as a means to coordinate ranging exchanges.

Our study exploits some of the findings in these works, but significantly differs from them in its grander goal to:

i) determine whether *different flavors* of concurrent transmissions can be embodied in a *full-fledged protocol and system* and, in the process,

ii) highlight *similarities and differences* w.r.t. their narrowband counterpart in terms of both *implementation complexity and system performance*, and ultimately

iii) provide a *reference implementation* of concurrent transmissions protocols that can be *directly* used and improved by the research community at large, fostering the adoption of this technique on UWB radios.

Which type of concurrent transmissions? As mentioned, several "flavors" of concurrent transmissions exist. Glossy

was originally designed to support a single network-wide flood, triggered at an *initiator* node; all nodes disseminate the *same* packet via a tightly synchronized schedule of alternating RX and TX slots, until the desired number N of packet (re)transmissions are performed.

In this paper, we consider other two variants, representative of the state of the art. On one hand, in the last editions of the above EWSN competition several systems [14, 1] achieved very high performance by changing Glossy to exploit only the single initial RX slot necessary to receive the packet, followed by N consecutive TX slots catering for its re-transmission. Given that the RX energy costs of the popular UWB platform we use are almost twice than TX ones, this TX-centric operation is definitely worth investigating. On the other hand, several works [10, 13] observed that floods are quite reliable even when *different* packets are concurrently transmitted by different initiators in the original Glossy scheme, offering another dimension to our study.

Methodology and contribution. Our investigation is system-driven, and relies on *complete protocol implementations* of the variants above as well as *testbed experiments* on multi-hop topologies. We use the popular DW1000, and specifically the EVB1000 boards, as our target UWB platform, and develop software atop ContikiOS, exploiting the availability of drivers for the DW1000 [3]. This methodology is in contrast with [19], whose results rely on microbenchmarks with few nodes in the same neighborhood, and also with [12], whose very short description of their Glossy-like component is insufficient to ascertain its actual performance or guide further developments.

This system-driven emphasis enables us to directly face the opportunities and challenges in exploiting UWB concurrent transmissions as well as to highlight key differences w.r.t. the corresponding narrowband implementations. One prominent example is the ability of the DW1000 radio to precisely schedule transmissions, which greatly simplifies implementation. Further, it also allows us to confirm, and sometimes disprove, some of the findings in [19, 12], ultimately contributing to a better system-level understanding of UWB concurrent transmissions.

We re-implemented Glossy and its TX-based variant from scratch, motivated by key differences in the radio operation and configuration. However, we also used the publicly-available codebase of Crystal [10, 11] a recent protocol that exploits both classic, single-initiator, same-packet floods as well as multiple-initiators, different-packet ones. As these are combined in a single protocol, Crystal serves as a sort of "catch-all" protocol enabling us to experiment with different types of concurrent transmissions in a single system. Further, the fact that we *reuse* the original Crystal codebase allows us to ascertain the extent to which this higher-level protocol built atop a narrowband Glossy layer can work when the latter is replaced with our UWB-based one. Our analysis shows that only minimal changes are required, suggesting that existing Contiki implementations of other higher-level protocols [7, 13, 16, 15] may be similarly reused for UWB radios, with minimal changes.

We illustrate the salient details of our implementations of the two Glossy variants (§4) and of Crystal (§5) hand-in-
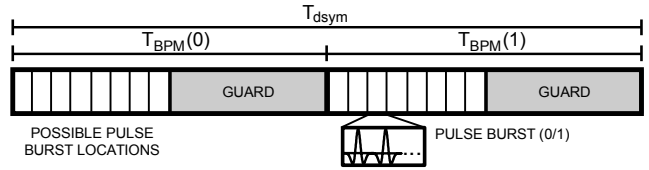

Figure 1. UWB data symbol.

hand with their evaluation in a 23-node indoor testbed at our premises, which enables us to experiment at scale on multi-hop topologies. Results show that UWB concurrent transmissions yield benefits similar to narrowband, achieving near-perfect reliability, and very low latency and energy consumption across the 4 hops in our testbed. Further, due to the high-accuracy clock and the high data rate, they also enable order-of-magnitude improvements in network-wide time synchronization. Nevertheless, as our experimental results are inevitably biased by the peculiarities of our testbed, we also manipulate artificially our setup to investigate the conditions under which UWB concurrent transmissions may fail, validating or disproving earlier findings [19, 12].

The paper ends by distilling findings and lessons learned (§6) that will hopefully inspire further work on the topic, before ending with brief concluding remarks (§7). We argue that our results pave the way for the exploitation of concurrent transmissions in UWB, which we foster by releasing our systems as open source[1], enabling their immediate use and improvement by researchers and practitioners.

2 Background

We provide a concise primer on the two main topics of this paper: UWB radios and concurrent transmissions.

2.1 Ultra-wideband Radios

Modern UWB radios are impulse-based, using carrier-less signals where information is encoded through short pulses ($\sim$ 1 ns). The IEEE 802.15.4 standard defines the structure of a UWB frame, the embedded error correction mechanisms and the decoding procedure. Here we briefly introduce the standard UWB physical layer and provide the most important details on the DecaWave DW1000 chip used in our evaluation.

UWB frame structure. The UWB frame is divided in two parts with different encodings: *i)* the synchronization header (SHR), and *ii)* the modulated portion embedding the data payload. The SHR is constructed from standard-defined preamble codes drawing from a ternary alphabet $\{+1, 0, -1\}$, corresponding to a positive, absent or negative pulse. The SHR ends with the start-of-frame delimiter (SFD), that indicates the beginning of data modulation. The IEEE 802.15.4-2015 standard [9] describes the sequence of steps for the creation of the UWB waveform to be transmitted for each radio configuration, starting from the input payload. After inserting forward error correction (FEC) bits in the payload and physical header (PHR), data undergoes convolution coding, modulation and time-hopping spreading.

BPM-BPSK data symbol. UWB uses a combination of burst position modulation (BPM) and binary phase-shift key-

---

ing (BPSK). Each BPM-BPSK data symbol (Figure 1) carries two bits of information, but only represents one input bit due to convolution coding. The data symbol of duration T_{dsym} is partitioned in two halves of duration T_{BPM}, where only one of the two halves is meant to host a burst (or pulse train). The number of pulses in the burst depends on the configured data rate. For the 6.8 Mbps rate used in this work, a burst is made of two pulses. If the radio transmits the pulse burst in the first half, it is interpreted as a 0 bit, 1 otherwise. The second bit is encoded by the phase (polarity) of said burst. Guard intervals, in which pulses are never transmitted, are placed between possible burst positions to serve as protection from high-energy multipath signal components, preventing inter-symbol interference. The combination of BPM and BPSK modulation schemes supports both coherent and noncoherent receivers, as the latter are unable to extract polarity information but can still decode based on the burst positions. To allow multi-user uncoordinated access, the location of pulses within a T_{BPM} duration is defined by a time-hopping sequence, derived from the preamble code.

DecaWave DW1000 and EVB1000. The DW1000 is a standard-compliant fully-coherent UWB transceiver, commercialized by DecaWave and included in the EVB1000 platform we use in our experiments. The DW1000 supports frequency channels $\{1–5, 7\}$, each with two different pulse repetition frequencies (*PRF*), nominally 16 MHz and 64 MHz. Three data rates are available: 110 kbps, 850 kbps, and 6.8 Mbps. The DW1000 requires an external 38.4 MHz oscillator, with a tolerance of ± 20 ppm [5]. This reference clock is used as phase-locked loop input to obtain a frequency of 125 MHz, allowing packets to be scheduled for delayed transmission with a precision of 8 ns. The programmer can trim the crystal frequency with a step that depends on the platform capacitors. In our case, the step is ~ 1.45 ppm.

Reception errors. UWB transmissions employ forward error correction in the form of convolution coding, combined with Reed-Solomon (RS) and SECDED bits (single error correction, double error detection) in the data payload and the PHR, respectively. Additionally, a 2-bytes CRC sequence is appended at the end of the frame. Uncorrectable bit errors are reported in the status register of the DW1000. The radio also signals SFD timeouts, that occur if the preamble of a frame is detected, but the presence of the SFD cannot be ascertained within the configured time interval.

2.2 Concurrent Transmissions

PHY-level enablers in narrowband. Protocols based on concurrent transmissions rely on two phenomena characteristic of IEEE 802.15.4 narrowband [2]. Constructive interference occurs when the *same* packet, transmitted by different senders, arrives at the receiver with a maximum time displacement of 0.5 μs, the duration of a bit (chip) in the transmitted chip sequence obtained by the direct-sequence spread spectrum (DSSS) encoding of the original message. The capture effect, instead, occurs even for *different* packets, as long as they arrive with a relative shift of no more than 160 μs, i.e., the duration of the synchronization header. In this case, one of the packets is likely received, depending on the density of neighbors and their relative signal strength.
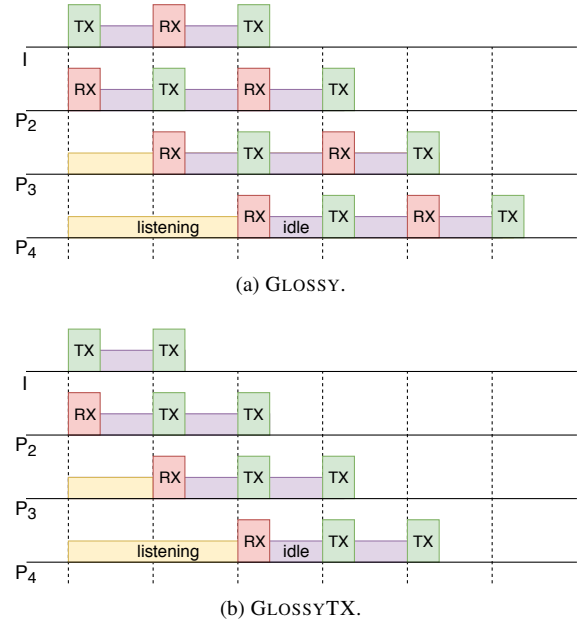

(a) GLOSSY.

(b) GLOSSYTX.

Figure 2. The two Glossy variants in a 4-hop network.

GLOSSY. Originally designed for multi-hop time synchronization, the Glossy [8] protocol exploits the two phenomena above to achieve fast, energy-efficient, and reliable network floods. Figure 2a illustrates the concept. The *initiator* begins a flood by broadcasting a packet. As the rest of the network is assumed to be already listening on the channel, the packet is received and immediately rebroadcast by neighbors, yielding concurrent transmissions. After (re)transmitting, the nodes go back to receiving, thus repeating the RX/TX sequence up to N times; the value of N is key to determine the balance between reliability and energy consumption. Another important factor affecting energy consumption is the duration of the slots, which must be long enough to accommodate either a packet TX or RX, including some guard times and software delays; nevertheless, when a packet is *not* received, a node listens for the entire slot, potentially wasting energy.

GLOSSYTX. A GLOSSY[2] flood unfolds by alternating RX and TX slots (Figure 2a); actually, a node is allowed to TX a packet *only* after a successful RX. This choice was originally motivated [8] by the use of CC2420 radio events as a means to enforce tight synchronization. However, it has drawbacks; a node that receives a packet in a RX slot and loses it in the next one is forbidden from rebroadcasting the (same!) packet in the subsequent TX slot, wasting time and energy, and possibly decreasing reliability. GLOSSY partially mitigates these problems by allowing only the initiator—i.e., the synchronization source—to transmit in a TX slot regardless of the outcome of RX ones, improving the flood progress in some unlucky situations.

Figure 2b shows an alternative scheme in which each node, after the initial successful RX, performs its N retrans-

---

[2]Hereafter, we use "Glossy" to refer generically to the system in [8], and "GLOSSY" to refer to the specific scheme derived from it (Figure 2) and implemented in this paper.
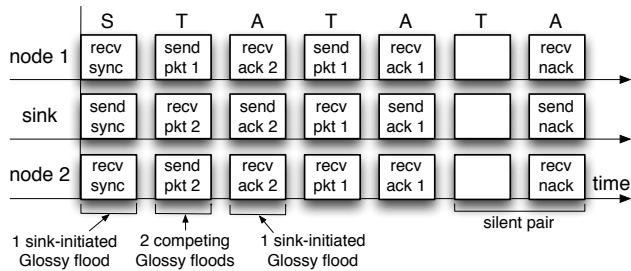
Figure 3. A Crystal epoch (from [11]).

missions in consecutive TX slots. This approach, hereafter called GLOSSYTX to distinguish it from the original, was first introduced by the winners of the 2nd EWSN Dependability competition [14], and exploited by other teams in following editions. A major drawback of GLOSSYTX is that its implementation, relying solely on timeouts, makes it more challenging to ensure tight synchronization of concurrent senders on TelosB-like devices. Further, more nodes transmit concurrently, possibly increasing the probability of collisions [2, 13]. On the other hand, GLOSSYTX unlocks several potential advantages by: *i)* solving the problem above induced by the original GLOSSY scheme, therefore potentially improving latency and/or reliability *ii)* enabling significant energy savings, by shortening the radio-on time by removing the unnecessary RX slots or, dually, *iii)* enabling reliability improvements, by replacing them instead with up to $N-1$ TX slots.

The fact that GLOSSY and GLOSSYTX strike different tradeoffs would already be enough motivation to consider them both in this paper. However, an even more compelling reason is the fact that, in the popular DW1000 UWB radio we use, the RX current draw is almost twice than the TX one, making GLOSSYTX preferable, at least in principle.

Higher-level Abstractions: Crystal. The effectiveness of Glossy gave rise to several protocols that are built directly atop the original implementation [7, 10] or slight modifications thereof [13, 16, 15]. Among these, Crystal is particularly suited for the study presented in this paper because *i)* it does not require modifications to Glossy, therefore allowing us to explore the extent to which the original narrowband can be replaced by our UWB implementations, and *ii)* it exploits concurrent Glossy floods containing *different* packets along with conventional, isolated ones, as described next.

Crystal [10] targets scenarios with aperiodic data collection and sparse traffic (e.g., those induced by data prediction, which provided the original motivation [10]) where relatively long periods of inactivity are interleaved with simultaneous data reporting from several nodes towards a sink. Catering for these scenarios requires a careful balance between the need to minimize energy consumption during the inactive periods and to guarantee timely and reliable delivery of data whenever needed. Crystal achieves this balance by *i)* dividing time into periods (*epochs*) that define the granularity of reporting, *ii)* exploiting the reliability of Glossy floods even when concurrently disseminating *different* pack-

ets, and *iii)* dynamically scheduling them as needed during the active time of the epoch, and putting the radio to sleep during the rest of it.

A Crystal schedule unfolds at the beginning of the epoch (Figure 3) and is composed of three phases, each corresponding to a Glossy flood:

- the initial S phase, ensuring time synchronization;
- the T phase, used by concurrent senders to disseminate their data. It is therefore the crucial phase, where *different packets* compete within concurrent Glossy floods originating from *different initiators*;
- the A phase, directed from the sink to all the nodes. It is performed in isolation and provides a network-wide acknowledgment of sorts, enabling each sender to determine whether a retransmission—another Glossy flood in the next T phase—is needed or not.

Termination occurs at each node when an empty T phase followed by an A phase containing no acknowledgment are observed for a number R of times.

3 Related Work

Concurrent transmissions, pioneered by Glossy, have been a breakthrough in narrowband low-power networking, showing unprecedented performance and leading to numerous follow-up works. It is therefore not surprising that researchers have begun investigating their applicability to other radio technologies, e.g., Bluetooth Low Energy (BLE) [17]. However, bringing techniques and results from narrowband to the impulse-radio UWB is non-trivial, due to the significantly different characteristics of the PHY layers.

A recent work [19], based on single-hop microbenchmarks, experimentally verified that concurrent transmissions are possible on UWB links. This holds with *identical* frames, but it was observed that also *different* frames can be supported under certain conditions, namely de-synchronization and signal strength disparity. These findings lie the foundation for our study, where we exploit concurrent transmissions in actual full-fledged systems for multi-hop data dissemination and collection. Moreover, to further investigate the limitations of the schemes we employ, we analyze the synchronization requirements for correct reception on the time scale of a single data symbol, unlike [19].

Another study [12] reported the use of a Glossy-like protocol to support a UWB localization system, i.e., the main contribution. The work described two necessary conditions for the correct operation of Glossy: preventing data symbol collisions and ensuring signal coherency of concurrent transmissions. As concurrent transmissions were not the main focus, however, the authors provide very few implementation details and no performance evaluation.

UWB concurrent transmissions have also been recently applied for concurrent ranging [4], in which all receivers of a single ranging request reply together. The authors show that the channel impulse response (CIR) available on the DW1000 can be exploited to collect multiple time-of-flight measurements at once. However, the system is not designed for communication, and the reliability of reception is only tested for the purpose of ranging in a single-hop scenario.

4 Glossy on UWB

We first illustrate our implementation of GLOSSY and GLOSSYTX on UWB (§4.1), focusing on how we exploit the opportunities offered by the DW1000 chip. We then quantitatively evaluate the performance of both variants (§4.2), drawing parallels with their narrowband counterparts. Finally, we analyze potential threats to the correct operation of our implementations and critically revisit some of the findings reported in the literature (§4.3).

4.1 Implementation Highlights

The original implementation of Glossy [8] targeted the TelosB motes (CC2420 radio and TI MSP430F1611 MCU) and was technically complex due to the lack of hardware support for precise timestamping of received packets and scheduling retransmissions. The clocks of the radio and the MCU run asynchronously, which causes a random jitter in the transfer of digital signals between these two components. Therefore, it is difficult to guarantee that the MCU issues the TX command to the radio at a designated time precisely enough, due to the non-deterministic time that elapses between a detected radio event and the invocation of the corresponding interrupt service routine (ISR). Moreover, the original Glossy avoided using the platform timers to schedule transmissions, because the stable 32kHz clock does not provide a sufficient resolution and the 4MHz DCO clock is not stable enough. Therefore, all actions of the protocol are triggered solely by radio events (e.g., *end-of-RX, end-of-TX, SFD*), further complicating the implementation.

The implementations described in this paper are for the DecaWave EVB1000 board, equipped with the DW1000 UWB radio and STM32F105 ARM Cortex M3 MCU. Other MCUs can be easily supported, however their clock speed and the data rate of SPI bus connecting MCU and radio can affect the timing of Glossy floods.

The DW1000 simplifies the Glossy implementation on many accounts. First and foremost, the DW1000 gives access to its internal clock; this can be used to *i)* timestamp received frames with sub-ns precision, and *ii)* schedule delayed frame TX with an 8 ns granularity. Both opportunities simplify the implementation tremendously. Random delays in ISR execution are no longer a problem, as the radio can be instructed to begin TX at an *exact* time in the future. Further, there is no jitter or non-determinism, because a single component—the radio—both timestamps the RX and triggers the TX using the *same* built-in clock.

Two variants of Glossy. As mentioned in §2.2, our GLOSSY and GLOSSYTX implementations have different purposes. GLOSSY is a faithful re-implementation of the original system in [8], where we exploit the precise timestamping and TX scheduling of the DW1000. Notably, we retain the original scheme in which a TX can be performed only for the packet received in the immediately preceding RX slot, except at the initiator (§2.2). This constraint was motivated in [8] by the need to obtain accurate timing information, and is made superfluous by the DW1000 features. Nevertheless, we preserve it to avoid changing the protocol too much, with the intent to have a yardstick enabling *direct* comparison with the body of literature on narrowband Glossy.

Table 1. Operation durations for UWB packets (μs).

	SHR	PHR + payload	SPI read + write	Other
15 B	73	45	∼36	∼250
127 B		178	∼304	

Table 2. Slot durations for UWB and NB (μs).

	UWB	NB
15 B	404	887
127 B	806	4471

Indeed, if one were to allow a TX of a received packet regardless of the outcome of the preceding RX slot, the purpose of the latter would become unclear. A more efficient protocol would be one where, after the first successful RX, the packet is transmitted *N* times without other RX slots. This is exactly what the GLOSSYTX variant does, for which our implementation takes full advantage of the DW1000 features. Direct access to the stable clock of the radio greatly simplifies implementation. The latter was actually the major hurdle pointed out by the literature [14, 1], which however lacks in-depth evaluations comparing GLOSSYTX vs. GLOSSY.

Anatomy and duration of a slot. A Glossy slot must account for the time necessary to: *i)* read/write the frame payload from/to the radio via SPI, *ii)* transmit/receive the frame synchronization header, *iii)* transmit/receive the physical layer header and the payload, *iv)* perform various software and hardware operations required for packet processing and radio configuration. Table 1 shows approximate durations of these steps in the EVB1000, for the two packet lengths we experiment with. By summing up the duration of all the steps, we obtain the actual slot sizes. Their comparison with corresponding slot sizes of narrowband Glossy (Table 2) shows a key advantage of UWB: the higher data rate (6.8Mbps vs. 250kbps on the CC2420) allows for slots that are 2.1x and 5.5x smaller, with evident benefits in latency.

On the other hand, concerning the first step above, the DW1000 does not support writing/reading the frame payload during its TX/RX, a feature of the CC2420 which increases parallelism. The DW1000 does allow uploading the payload in parallel with transmitting the preamble; however, we could not exploit this feature because, for the preamble setting we used, the former is slower than the latter.

Dynamic clock frequency calibration. The radio clock of our platform is very stable. Even though DW1000 tolerates up to ±20 ppm [5] frequency drift (§2.1), the EVB1000 platform we use integrates a ±10 ppm oscillator, individually calibrated (trimmed) by the manufacturer to achieve ±3 ppm in normal conditions. Nevertheless, temperature and voltage variations may cause its frequency to drift within the full ±10 ppm range.

The authors of [12] report that this drift may undermine the reliability of concurrent transmissions. Therefore, we implement a dynamic frequency calibration of the radio clock of the receivers, relative to that of the flood initiator. Inspired by [12], the calibration is achieved by observing the time offset between the expected and actual arrival of consecutive floods, and adjusting the radio oscilla-
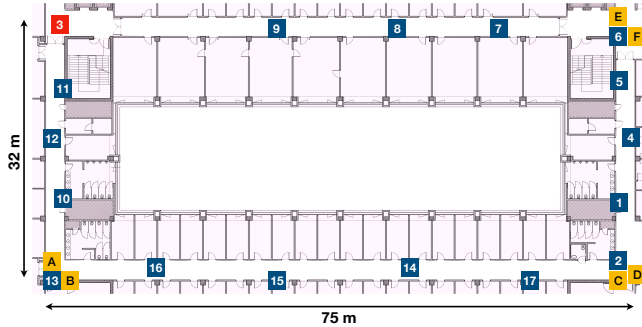
Figure 4. Experimental testbed. Out of the 23 nodes available, 22 were running the protocols under study; node 3 served as a sniffer.

Figure 5. Latency of node 11, the farthest from the initiator (4 hops). Bars denote minimum/maximum values; boxes denote the 25–75% percentile.

tor frequency of every receiver appropriately. This is done by trimming the oscillator with a hardware-defined step of 1.45 ppm. By choosing the value closest to the desired frequency, we ensure that the frequency offset of any device w.r.t. the flood initiator is within ± 0.725 ppm. For any pair of non-initiator devices, their relative frequency offset stays within 1.45 ppm.

This dynamic calibration requires the radio clock to remain active in between floods, with the radio in idle mode. This has relatively high power consumption (Table 4); however, the calibration is in general infrequent. In §4.3 we further elaborate on the impact of this technique via experiments that provide additional insights beyond what reported in [12], whose results are based on a custom hardware design achieving higher synchronization accuracy.

4.2 Evaluation

We evaluate several aspects of GLOSSY and GLOSSYTX, highlighting similarities and differences between them and w.r.t. their narrowband counterparts.

Experimental setup and radio configuration. We tested our implementation of Glossy in a 23-node testbed deployed in the corridors of an office building (Figure 4). The communication range normally extends through the entire length of each straight segment, with at least one link with $PRR \geq 90\%$ for every pair of adjacent corners. However, exceptions exist where shorter links are less reliable, e.g., node 17 cannot communicate directly with 13. Further, we verified that node 11 cannot communicate directly with 9; therefore, we do not use the node 3 in between them, and set node 9 to be the initiator, achieving a network diameter of 4 hops.

As for the radio configuration, we use channel 4, 6.8 Mbps data rate, 64 MHz *PRF*, 64 *μs* preamble, and the maximum transmission power of `0x9A9A9A9A` recommended [6] for the combination of channel and *PRF* we use.

Flood reliability. One of the main benefits of concurrent transmissions is their ability to achieve near-perfect reliability. The latter strongly depends on the number N of retransmissions (§2.2). However, long packets are also known to be detrimental to reliability in narrowband [8]. For these reasons, we experiment with $N \in \{1, 2, 4, 8\}$ and *i)* short packets of 15 B, allowing for 8 B of payload as commonly used in the literature, and *ii)* long packets of 127 B, the maximum
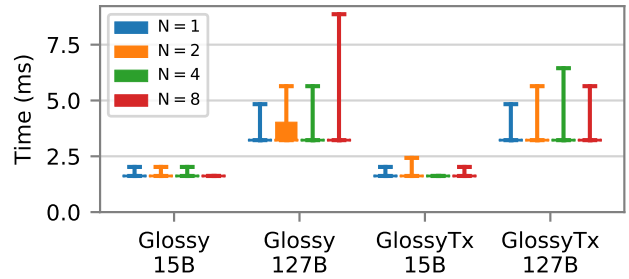
allowed by the standard. For every combination of these values, we report results aggregated from 12000 floods.

Table 3 shows that, in our experiments, both variants always achieved perfect reliability with short packets, even with N=1. Instead, with long packets this is achieved only by GLOSSY and only with N=8; further, GLOSSYTX is systematically less reliable for $N \geq 2$, although it achieves a reliability $\geq 99\%$ in all cases. Interestingly, the reliability of GLOSSY increases with N, as expected, while this is not always true for GLOSSYTX. For $N \geq 2$, in GLOSSYTX the number of TX slots at each node is higher than RX slots, increasing the number of nodes transmitting simultaneously and therefore the chance of occasional collisions among the (long) packets.

Latency. These trends are mirrored by the *first relay count*, i.e., the number of slots elapsed at a node before the first successful RX slot, effectively an indirect measure of latency. The values for this metric are identical for the two variants in the case of short packets or N=1, but are slightly higher in the other cases, meaning that the flood is slightly delayed due to lost packets and consequent retransmissions.

To investigate the *maximum* latency, Figure 5 focuses on node 11, the farthest from the initiator. In most cases, the flood reaches this node exactly after 4 hops, with sporadic outliers in case of short packets. With long packets, the maximum latency is still very stable, though bigger, due to larger Glossy slots needed. However, the 99th percentile shows increase in latency corresponding to 1–2 slots. Overall, there is a weak tendency for latency to grow when N increases, because of a higher chance of collision, as discussed before.

Compared to narrowband [8], our UWB implementation provides smaller latency due to the shorter slots used, with a 52% reduction for short packets and 82% for long packets.

Energy consumption. Unlike narrowband radios like the CC2420, for which TX and RX have similar energy costs, the RX current draw of the DW1000 chip is almost twice than the TX one (Table 4). This motivates investigating the energy consumption of the two Glossy variants, as they exploit very differently these two radio states. However, this energy unbalance prevents us from using radio-on time as an energy metric, as commonly done by the narrowband literature. We therefore resort to *modeling* directly the energy costs, as the structure of Glossy protocols is simple and largely determin-

Table 3. GLOSSY vs. GLOSSYTX: reliability and latency.

	Frame length (bytes)	Average flood reliability, %				Minimum node reliability, %				Mean first relay count			
		$N=1$	$N=2$	$N=4$	$N=8$	$N=1$	$N=2$	$N=4$	$N=8$	$N=1$	$N=2$	$N=4$	$N=8$
GLOSSY	15	100	100	100	100	100	100	100	100	1.32	1.32	1.32	1.32
	127	99.91	99.997	99.9992	100	99.5	99.95	99.991	100	1.49	1.37	1.46	1.49
GLOSSYTX	15	100	100	100	100	100	100	100	100	1.32	1.32	1.32	1.32
	127	99.91	99.97	99.95	99.997	99.5	99.8	99.0	99.95	1.49	1.48	1.56	1.46

istic. Specifically, we study the energy cost of GLOSSY and GLOSSYTX as a function of the hop distance from the initiator; however, we neglect the contribution of collisions, as these are generally rare and in any case dependent on the specific target environment and network topology.

The drain of electric charge of a node during a flood is:

$$Q = T_{TX}I_{TX} + T_{RX}I_{RX} + T_{listen}I_{listen} + T_{idle}I_{idle} \quad (1)$$

where T_i and I_i are, respectively, the time spent and corresponding current draw in a given state i (Figure 2). From this, energy can be computed easily as $E = Q \cdot V$, with knowledge of the voltage supply (Table 4).

The current draw for each state is shown in Table 4 for both the DW1000 and the CC2420 used in the original implementation of Glossy. In the latter case, the values are retrieved from the datasheet [18]. For the DW1000, the datasheet does not contain information specific to the radio configuration of our experiments; therefore, we use the current draw reported for the most similar one, i.e., the one with the same parameters but channel 2, which has the same central frequency of channel 4 but smaller bandwidth. Note that radio states with a lower power than the idle one cannot be exploited, due to the large time required by the radio to exit from them (e.g., up to $\sim$3 ms for the DW1000).

Interestingly, (1) also models the consumption of *narrowband* GLOSSY and GLOSSYTX, albeit with a few caveats. Indeed, the original implementation for CC2420 reads and writes frame data via SPI directly during RX and TX, respectively, avoiding inter-slot processing delays and the need for putting the radio to the idle state between RX/TX slots. Therefore, to apply our energy model (1) to narrowband, we *i)* consider as T_{idle} the time (192 μs) needed by the radio to switch from RX to TX and the software delay (23.3 μs) required by the MCU to trigger a TX, and *ii)* account for it as if the radio were in RX. This, along with the fact that $I_{listen} = I_{RX}$ for narrowband (Table 4) leads to the simpler expression for *narrowband* variants:

$$Q = T_{TX}I_{TX} + (T_{RX} + T_{listen} + T_{idle})I_{RX} \quad (2)$$

The values of T_i can instead be determined as a function of N, of the slot duration (T_{slot}), of the radio time to TX or RX a frame (T_{frame}), and of the first relay counter (C):

$$
\begin{aligned}
T_{TX} &= N \cdot T_{frame} \\
T_{RX} &= N_{RX} \cdot T_{frame} \\
T_{listen} &= C \cdot T_{slot} \\
T_{idle} &= (N_{RX} + N - 1) \cdot (T_{slot} - T_{frame})
\end{aligned}
\quad (3)
$$

where $N_{RX} = N$ for GLOSSY, and $N_{RX} = 1$ for GLOSSYTX.

Figure 6 shows the resulting energy estimates for $N=4$; different N values exhibit similar trends. As expected,

Table 4. Nominal current draw and voltage supply.

	frame size (B)	current draw (mA)				voltage (V)
		I_{RX}	I_{listen}	I_{TX}	I_{idle}	
CC2420	any	18.8	18.8	17.4	0.426	3.0
DW1000	15	114.9	113.0	71.5	18.0	3.3
	127	116.5	113.0	61.1	18.0	

--- NB-Glossy   -- NB-GlossyTx   — UWB-Glossy   — UWB-GlossyTx
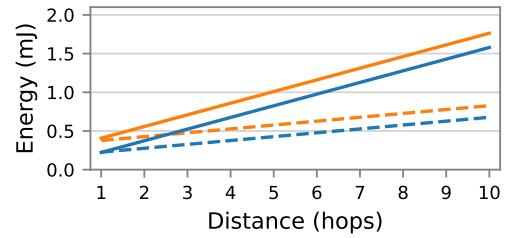
(a) 15B frame.
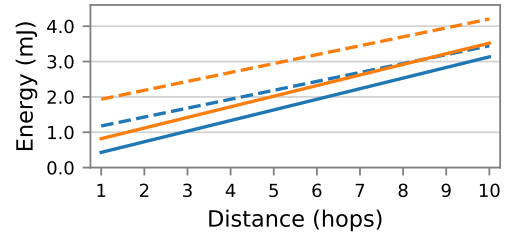
(b) 127B frame.

Figure 6. Energy consumption: GLOSSY vs. GLOSSYTX for narrowband (NB) and UWB. Note the difference in scale between the *y*-axes.

GLOSSYTX is more energy-efficient than GLOSSY both in narrowband and UWB. By scheduling only TX slots after the first successful RX, GLOSSYTX reduces the flood duration, sparing energy. This difference increases with N. As for the tradeoffs between narrowband and UWB, with short packets the former clearly outperform the latter. Interestingly, roles are reversed when long packets are transmitted. Despite the higher energy cost of both TX and RX for the DW1000 chip (Table 4), UWB GLOSSYTX is the most efficient solution. At the first hop it consumes nearly one third of its narrowband counterpart, and 4.5x less than narrowband GLOSSY. However, the gap decreases with hop distance.

The reason behind the higher energy efficiency of UWB-based solutions with larger payloads is twofold: *i)* the data rate of CC2420 is 27x smaller than DW1000 (250 kbps vs 6.8 Mbps); to TX (or RX) 127B of data, narrowband radios stay active $\sim$18x longer than UWB *ii)* the processing delays
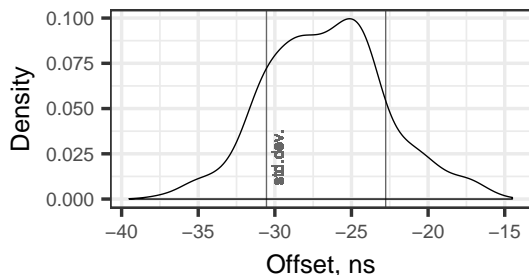
Figure 7. Time synchronization error.

Figure 8. *PRR* **when a transmission is shifted over the symbol duration (short 15 B frame).**

required by the UWB implementation do not bear a significant negative impact on energy consumption as the radio remains idle; after the first successful RX, nodes spend in idle >60% of the time, saving considerable energy.

Accuracy of time synchronization. Finally, we recall that Glossy was originally proposed for time synchronization [8]. It is therefore interesting to investigate this aspect, especially given that the UWB platform we use provides access to its highly accurate clock. To study the synchronization accuracy, we *i)* rely on the privileged position of node 3 and use it as a "sniffer", capable to hear and timestamp the RX of packets sent by node 9, the initiator, and node 11, the farthest from it, and *ii)* analyze their difference w.r.t. the first TX of the initiator (relay count of 0) and the first TX of node 11 (relay counter of 4).

As Glossy was, by design, unaware of the distance among nodes, the reference time at the receivers is always biased by the signal propagation delay, ∼333 ns every 100 m. Knowing the overall distance the signal travels in our setup, we subtract that bias and determine the error distribution. This yields a setup similar to the original in [8], where nodes were all on the same desk and propagation time essentially negligible. Results show an underestimation of ∼6.5 ns per hop, resulting in an average offset of −26 ns at 4 hops (Figure 7). We attribute this bias to an imprecise antenna delay calibration. Overall, the error distribution covers an interval of 22 ns, essentially due to the 8-ns precision in the DW1000 TX scheduling, accumulating over 4 hops. In the worst case, the TX scheduling error is always exactly 8 ns, yielding a theoretical maximum error of 32 ns for our setup; in practice, the random variations of TX times often cancel each other. In any case, the standard deviation of the error is 3.89 ns, i.e., *almost three orders of magnitude smaller than in narrowband*, reported in [8] to be 2.5 μs over 4 hops.

4.3 Exploring the Limits

As we observe instances of frame interference in our testbed setup, we investigate the conditions that can hamper concurrent transmissions in our UWB implementations of Glossy. To this end, we collect empirical evidence in a different, smaller-scale setup where we can precisely control the overlapping of signals. We position a receiver in between 2 synchronized transmitters, at 1 m distance from each, and evaluate the effect of data symbol misalignment and crystal accuracy. This placement is particularly challenging because the strength of concurrent signals is similar at the re-
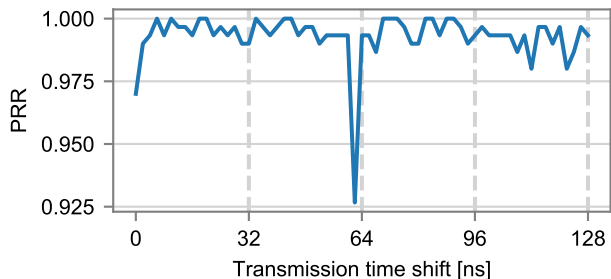
ceiver. However, it allows us to derive stronger conclusions regarding the limitations and the ideal conditions of concurrent transmissions, as well as provide evidence that the signal strength and the number of available receivers play a role in the robustness of Glossy.

Payload collisions. UWB data symbols are divided in two halves for binary burst position modulation (BPM, §2.1). In principle, a node may fail to receive correctly when two concurrent transmissions are shifted by more than half a symbol duration, and one of the pulse bursts occupies the wrong BPM location. The ability to ensure that different transmissions occupy the same BPM locations has been reported as a *necessary* condition to prevent collisions [12]. Specifically, pulse bursts should remain within the T_{BPM} duration (§2.1), i.e., 64.105 ns for the 6.8 Mbps data rate used in our setup, resulting in severe limitations on the position of nodes.

However, another related work [19] and our earlier experiments—neither of which mentioned this constraint— hint at the fact that this is actually not at all crucial. To verify this hypothesis, we apply a correction to schedule TXs with a precision of ∼ 1 ns (instead of 8 ns) and delay one of the transmitters to cause different degrees of signal overlapping along the symbol duration. Figure 8 shows that the receiver enjoys a *PRR* ≥ 98% even when the delay we artificially introduce is > 64.105 ns, causing pulses to occupy the opposite side of the data symbol. This proves that the DW1000 radio is able to decode the packet correctly even in the presence of pulse bursts in erroneous locations. We speculate that this is due to the coherency mechanisms built in the DW1000 receiver, that allow correct decoding based on the phase of the signal alone. On the other hand, Figure 8 also shows that the decoder is affected by concurrency when pulses are really close to ∼ 64 ns, i.e., when they occupy a location matching the time-hopping sequence on the opposite side of the BPM-BPSK symbol. However, this constraint is unlikely to happen and even less likely to disrupt a Glossy flood, thanks to spatial diversity and inherent variations of TX times.

Our findings significantly relax previously reported requirements [12] that, by indirectly affecting the physical location of network nodes, would otherwise hamper the practical applicability of concurrent transmissions over UWB.

Frequency offset. Another necessary requirement reported in [12] is about coherency of two overlapping signals throughout the whole frame. In other words, the phase drift
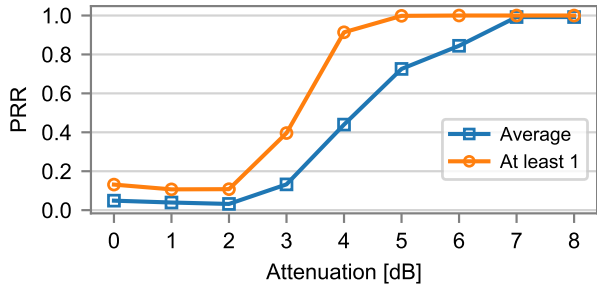
Figure 9. Reliability with 4 receivers and 2 transmitters with relative TX attenuation (long packets): average and at-least-one *PRR*.

caused by the oscillator frequency difference of two concurrent transmitters should never go beyond half the oscillation period within the frame transmission. In [12], this is translated into a maximum clock frequency offset of 1.39 ppm for 33B frames. The same calculation, applied to the maximum-length packets of 127B allowed by the standard and considered in §4.2, yields $0.5/3.4944 \times 10^9/252 \times 10^{-6} = 0.57$ ppm, where 252 μs is the packet TX time with the 64 μs preamble we use. This value is quite far from the 1.45 ppm we achieve with frequency calibration (§4.1). Indeed, in our controlled (and challenging) small-scale experiments above, we observe a rather low *PRR* (0–10%) regardless of whether the calibration is used or not. Unfortunately, further improvements are possible only with a custom hardware platform, different from the popular EVB1000 we use in this paper.

On the other hand, to elicit further insights about this constraint as well as assess its impact on short packets, we use a small-scale setup where we concurrently transmit 12200 rounds of short packets and simulate the presence of a transmitter with poor crystal accuracy by artificially altering the oscillator frequency, albeit within the ± 20 ppm tolerance required by the DW1000 [5]. First of all, we confirm that frequency offsets is not a problem in the case of *isolated* transmissions. In our experiments, a single transmitter with the artificial frequency offset of 10 ppm yields $PRR \geq 99\%$, as expected given that this offset is within the DW1000 tolerance. However, when *multiple concurrent* transmitters are present and one is configured with the same artificial 10 ppm offset, we obtain $PRR \geq 81.17\%$, while dynamic frequency calibration (§4.1) yields $PRR \geq 96.74\%$. This confirms that *i)* the frequency offset matters even for short packets, and *ii)* dynamic frequency calibration is effective in improving reliability.

On the other hand, we observed these drops in reliability only when *artificially* introducing a frequency offset, as witnessed by the perfect reliability shown in §4.2 for both Glossy variants. In practice, the EVB1000 platform we use, factory-trimmed at 3 ppm, guarantees perfect reliability even without dynamic frequency calibration. However, the latter may play a role in deployment environments harsher than the indoor one where we performed our experiments.

Receiver redundancy and TX power. The number of available receivers at a given hop and the relative TX power of transmitters are important factors in the reliability of our UWB Glossy variants, similarly to the narrowband ones [2]. This aspect, largely neglected by related work [19, 12], would deserve a more exhaustive analysis than what possible here. Nevertheless, we offer empirical evidence about it in our small-scale setup with 2 co-located nodes transmitting long packets—the most unreliable—and 4 receivers, again in the most challenging placement where they are at essentially the same distance from transmitters. Further, we configure the transmitters with a relative TX attenuation of 0–8 dB. Figure 9 shows that when the transmitters use the same TX power, the average *PRR* is very low; nevertheless, it nearly doubles if computed by considering a reception successful when it occurs on *at least one* of the 4 receivers. Further, it also shows that the *PRR* rapidly grows with the difference in TX power. An attenuation of 7 dB ensures that *all* receivers get the packet; 5 dB are sufficient to ensure reception by at least one of them. These considerations are important, as one successful receiver is enough to enable a Glossy flood to progress. On the other hand, the dual also holds; a topology in which progress is ensured by a single forwarder is obviously very brittle. Incidentally, this is the reason why we placed nodes $A - F$ in the corners of our testbed, that is, to eliminate these single-receiver bottlenecks that should anyway be avoided in real deployments.

5 Crystal on UWB

The previous section confirmed that our implementations of Glossy for UWB radios provide benefits comparable to those known from the narrowband literature. We now turn our attention to a different research question, namely, whether the results from higher-level abstractions and protocols built atop the Glossy layer also transfer to UWB. To provide an answer, we focus on the Crystal protocol [10, 11] described in §2.2. We discuss the few changes our Crystal implementation for UWB required w.r.t. the original one, followed by the results of its evaluation in our testbed.

5.1 Implementation Highlights

We used the publicly-available code for Crystal, and kept the overall protocol logic unchanged; we disabled channel hopping [11], as it is not the focus of this paper. However, a few minor modifications were necessary, motivated by the different operation of the underlying radios.

Crystal detects termination based on the absence of received packets; it is therefore crucial to tell apart absent transmissions from failed receptions. In narrowband, noise detection enabled Crystal to defer termination if no packet is received but strong noise is detected. However, this mechanism relied on clear-channel assessment (CCA), not present in UWB. Further, it did not provide direct evidence of a failed RX, but only of the *possibility* of one, due to noise.

On the other hand, the DW1000 offers rich information about RX errors, which we exploit in our implementation. This information is signaled when the radio detects a preamble but fails to decode either the SFD or the data portion of the packet, due to Reed-Solomon, SECDED or CRC errors (§2.1). A "spontaneous" preamble detection may still happen without any TX, but is highly unlikely in practice. On the other hand, the mere presence of a preamble signals that

one or more nodes are sending packets but their data cannot be decoded, probably because of collisions. We verified that these techniques significantly improve the reliability of our UWB implementation of Crystal.

Finally, our UWB implementation relies on the MCU 32 kHz timer of the EVB1000 board to schedule its activities; data TX in the shared T slots may therefore overlap within 30μs. Relying on the more accurate radio clock would require significant changes in the code, in contrast with our desire to minimize them. Further, it would likely bring little or no benefits, given that a slight de-synchronisation of transmitters is reported in [19] to increase the reliability of concurrent transmissions of *different* packets. Follow another recommendation of [19], we also increase the SFD timeout by 32 μs, to account for possible frame offsets caused by the timer resolution.

5.2 Evaluation

We evaluate Crystal in our testbed, with the same configuration of §4.2; further, node 9 is the sink, maximizing the network diameter. We are interested in the overall reliability in delivering packets at the sink, but also in ascertaining the underlying raw reliability of the floods disseminating *different* packets and competing in the same shared Crystal slot. In doing so, we experiment with both variants of the underlying Glossy, as well as with short and long packets.

We consider two key parameters influencing performance: the number U of concurrent updates and the number N of TX in the Glossy flood inside a shared Crystal slot. U determines how many nodes transmit a packet in each Crystal epoch, defining the degree of concurrency. We explore $U \in \{1, 2, 5, 7, 10\}$, i.e., up to half of the network, chosen at random among non-sink nodes; we also consider the extreme case where all the U=21 non-sink nodes transmit concurrently. N defines the degree of redundancy at the Glossy layer; we explore $N \in \{1, 2, 4, 8\}$ as in §4.2. We set a default of U=5 and N=2 when exploring the other parameter.

For each combination in this space we collect traces of 1000 Crystal epochs, i.e., $1000 \times U$ packets transmitted.

Overall reliability. Our UWB implementation of Crystal ensures remarkable reliability with both short and long packets. With the short 15B packets, Crystal correctly delivered *all* the > 150000 packets transmitted, regardless of the parameter configuration, and notably even when U=10 nodes (half of the network) were concurrently transmitting. Moreover, even with the longest 127B packets Crystal achieves >99.9% reliability. This near-perfect reliability is fully in line with the results originally reported in [10], therefore confirming that the performance Crystal achieves in narrowband can be harvested also in UWB.

Reliability of shared slots. This remarkably high reliability is achieved in Crystal via mechanisms that *mask* the packet losses in the underlying concurrent Glossy floods. Therefore, in the light of ascertaining the extent to which concurrent transmissions of *different* packets can be used as a building block for other protocols [7, 13, 16], we now focus on the performance of shared slots in isolation.

Specifically, we look at the first T slot of each Crystal epoch, when there are exactly U nodes transmitting simultaneously. For each node, we define the success rate metric as the ratio between the number of floods when the node received *any* packet over the total number of floods when the node was listening in the first T slot, as in [10]. Figure 10 shows the average success rate, and Figure 11–12 the distribution of this metric across nodes via the complementary empirical cumulative distribution function (CCDF).

The charts exhibit clear trends. First of all, GLOSSY systematically outperforms GLOSSYTX across all configurations. In particular, the reliability of GLOSSY remains nearly constant w.r.t. the increase in the number U of concurrent senders, while GLOSSYTX shows a marked decline. This is a consequence of the fact, already pointed out in §4.2, that the density of transmissions in GLOSSYTX is much higher than in GLOSSY, and obviously exacerbated as U increases. Long frames degrade reliability of both variants, again consistently with §4.2, and with a more marked effect on GLOSSYTX, as per the observations above. In any case, the absolute worse success rate recorded across all these many experiments was 95%, which is still very good.

Finally, reliability increases with the degree of redundancy induced by N. At the highest value tested, N=8, the average reliability of GLOSSY reaches 99.994% with short packets and 99.96% with long ones, caused by packet losses at a single node. As for GLOSSYTX, the average reaches a plateau of 99.7% for short packets and 99.0% with long ones. These figures are remarkable, considering that *i)* they are achieved with concurrent transmissions of *different* packets, and *ii) without* the reliability mechanisms of Crystal, which are nonetheless key to spare the steep energy costs induced by a high value of N.

Extreme case. We also tested Crystal in the extreme scenario where *all* 21 non-sink nodes transmitted in all epochs (U=21). With N=2, the success rate of the T phase drops as low as 80% even with short packets; nevertheless, the overall reliability at the sink remains at 99%. As expected, long packets exhibit worse reliability, with a success rate of the T phase at ∼75% and an overall reliability of ∼95%. At the other extreme, N=8 yields a near-perfect overall reliability for short packets, with several runs at 100%, and ∼98% for long packets, despite an underlying success rate at ∼88% and ∼79%, respectively. Overall, these results confirm once again the effectiveness of the "safety net" provided by Crystal reliability mechanisms, as already observed for narrowband in similar extreme scenarios [10, 11].

6 Discussion

We distill salient findings from our results and offer some considerations that may inspire future work on the topic.

Similarities vs. differences. First and foremost, our experiments demonstrate that Glossy-like mechanisms achieve in UWB benefits similar to narrowband, i.e., low latency, high reliability, low energy consumption——all at once. Energy consumption has slightly different tradeoffs than in narrowband, due to the significant imbalance between RX and TX in the DW1000. On the other hand, the latency achievable in UWB is significantly lower than in narrowband, as a consequence of the high-accuracy clock and the higher data rate. The former also enables *a three order of magnitude improve-*
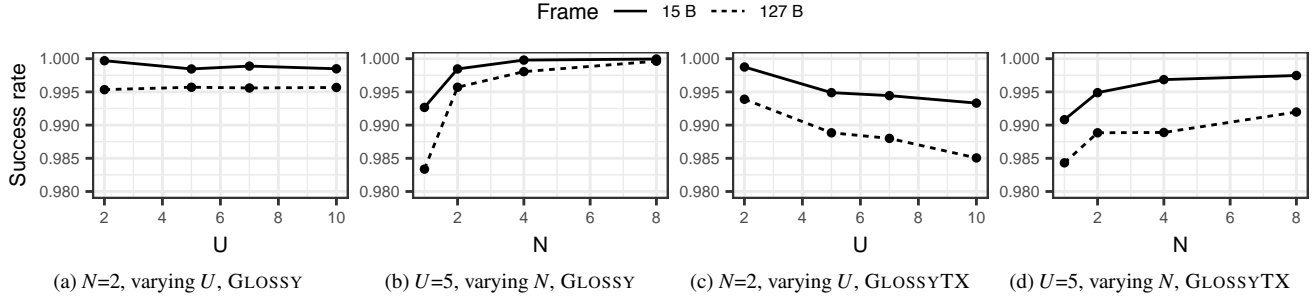
Figure 10. Average success rate in the first T shared slot, for different values of N and U.
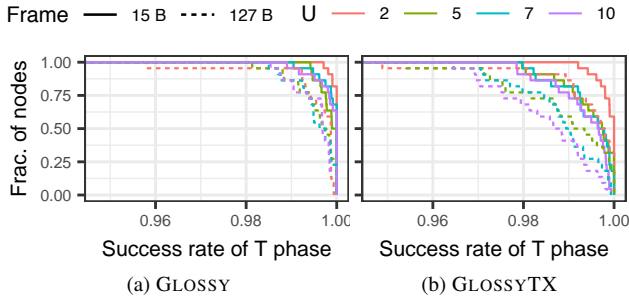
(a) N=2, varying U, GLOSSY   (b) U=5, varying N, GLOSSY   (c) N=2, varying U, GLOSSYTX   (d) U=5, varying N, GLOSSYTX

Figure 11. CCDF for T success rate vs. U (N=2).

(a) GLOSSY   (b) GLOSSYTX
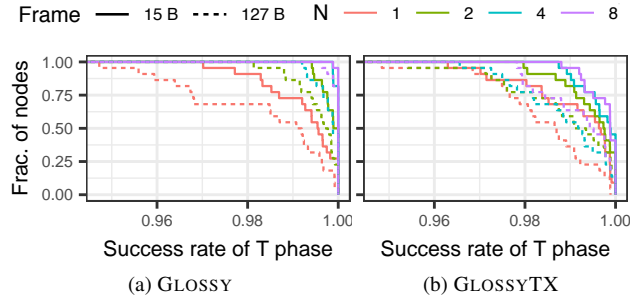
Figure 12. CCDF for T success rate vs. N (U=5).

(a) GLOSSY   (b) GLOSSYTX

ment in the accuracy of network-wide time synchronization, which was actually the original motivation for Glossy.

Interestingly, our in-depth analysis of §4.3 reveals that the key PHY-level mechanism enabling Glossy appears to be more *non-destructive interference* than a constructive one. Indeed, the DW1000 is capable of decoding concurrent pulses—and therefore packet TX—even when severely misaligned. On the other hand, due to the encoding based on short pulses rather than long waves, the crystal frequency offset is more of an issue for UWB than it is for narrowband, with a stronger impact on the tradeoff between the packet length of the concurrent transmissions and their reliability. However, as for narrowband, a difference in the signal energy of concurrent transmissions and multiple receivers improve performance considerably even for long frames.

GLOSSY or GLOSSYTX? A first observation is that the question is actually an open one for narrowband. Indeed, the potential superiority of GLOSSYTX is rather anecdotal, as it derives from the ad hoc setup of the EWSN Dependability Competition and has never been rigorously analyzed across different system parameters.

For UWB, our study shows that GLOSSYTX is more energy-efficient than GLOSSY; for long packets, it achieves a consumption even lower than narrowband. Therefore, it would seem obvious to always use it in place of GLOSSY. Nevertheless, our study shows that this is not always necessarily the case. Indeed, the other side of the coin is that the aggressive re-transmission policy of GLOSSYTX is prone to increasing the number of collisions, affecting reliability. This behaviour is notable both in same-packet and different-packet floods. However, the actual impact ultimately depends on how the Glossy layer is used in the specific traffic profile and/or higher-level system (e.g., Crystal, in our case).

The dual argument is that GLOSSY appears slightly more reliable, due to the alternating pattern of TX and RX slots that reduces the "density" of concurrent senders and thus the probability of collisions. An opportunity for future work is to find a scheme striking the right balance between the back-to-back transmissions of GLOSSYTX and the sparser transmissions, yet rigidly alternating with receptions, of GLOSSY.

Transferring results from narrowband to UWB. As repeatedly mentioned, there is a substantial literature on concurrent transmissions for narrowband, including systems that built atop the original Glossy to support alternate network functionality [7, 13, 10, 16].

Our experience with "porting" Crystal to UWB and the related evaluation (§5) shows that the effort required is relatively small while, on the other hand, the benefits that can be attained are entirely in line with those shown for narrowband. Of course, it would be a leap of faith to claim that the same can be done for all other higher-level abstractions in the literature. However, our experience hints at the fact that this may actually be the case for several of them, especially those that run atop an unmodified Glossy layer, e.g., notably including LWB [7].

We argue that pursuing this question is actually important, to amplify the impact (and awareness of) the body of literature on concurrent transmissions on other radio technologies and hence research communities, and concretely demonstrate that it applies to a far more general scope than the hardware niche it was originally developed for.

7 Conclusions

We explored the extent to which concurrent transmissions, made popular by Glossy for IEEE 802.15.4 narrowband, can be exploited in UWB via a full-fledged, readily-available system, and ascertained what is the corresponding

performance. Overall, the answer is very positive: both variants of Glossy we consider, as well as Crystal, the higher-level abstraction building atop of it, yield in UWB a reliability similar to the one observed in narrowband. Further, the higher clock accuracy and data rates in UWB unlocks significant latency improvements, which become order-of-magnitude ones for time synchronization, the original motivation of Glossy. We provided a detailed account of the opportunities this UWB platform enables for an efficient implementation of concurrent transmissions, an analysis of the threats to performance, as well as investigated the effort required to exploit Crystal atop the Glossy layer.

Beyond the qualitative lessons learned and quantitative results reported here, we also release the systems we described as open source, enabling their immediate use and improvement by researchers and practitioners, and generally inspiring future work on the topic.

8   Acknowledgments

9   References

[1] M. Baddeley, A. Stanoev, U. Raza, M. Sooriyabandara, and Y. Jin. Competition: Adaptive software defined scheduling of low power wireless networks. In *Proc. of EWSN*, 2019.

[2] T. Chang, T. Watteyne, X. Vilajosana, and P. H. Gomes. Constructive interference in 802.15.4: A tutorial. *IEEE Communications Surveys Tutorials*, 21(1):217–237, Firstquarter 2019.

[3] P. Corbalán, T. Istomin, and G. P. Picco. Poster: Enabling Contiki on Ultra-wideband Radios. In *Proc. of EWSN*, 2018.

[4] P. Corbalán and G. P. Picco. Concurrent Ranging in Ultra-wideband Radios: Experimental Evidence, Challenges, and Opportunities. In *Proc. of EWSN*, 2018.

[5] DecaWave Ltd. DW1000 Data Sheet, version 2.19, 2017.

[6] DecaWave Ltd. DW1000 User Manual, version 2.18, 2017.

[7] F. Ferrari, M. Zimmerling, L. Mottola, and L. Thiele. Low-power Wireless Bus. In *Proc. of SenSys*, 2012.

[8] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. Efficient Network Flooding and Time Synchronization with Glossy. In *Proc. of IPSN*, 2011.

[9] IEEE 802.15.4-2015, Standard for Low-Rate Wireless Networks.

[10] T. Istomin, A. L. Murphy, G. P. Picco, and U. Raza. Data Prediction + Synchronous Transmissions = Ultra-low Power Wireless Sensor Networks. In *Proc. of SenSys*, 2016.

[11] T. Istomin, M. Trobinger, A. L. Murphy, and G. P. Picco. Interference-Resilient Ultra-Low Power Aperiodic Data Collection. In *Proc. of IPSN*, 2018.

[12] B. Kempke, P. Pannuto, B. Campbell, and P. Dutta. SurePoint: Exploiting Ultra Wideband Flooding and Diversity to Provide Robust, Scalable, High-Fidelity Indoor Localization. In *Proc. of SenSys*, 2016.

[13] O. Landsiedel, F. Ferrari, and M. Zimmerling. Chaos: Versatile and Efficient All-to-all Data Sharing and In-network Processing at Scale. In *Proc. of SenSys*, 2013.

[14] R. Lim, R. Da Forno, F. Sutton, and L. Thiele. Competition: Robust flooding using back-to-back synchronous transmissions with channel-hopping. In *Proc. EWSN*, 2017.

[15] M. Mohammad and M. C. Chan. Codecast: Supporting Data Driven In-Network Processing for Low-Power Wireless Sensor Networks. In *Proc. of IPSN*, 2018.

[16] B. A. Nahas, S. Duquennoy, and O. Landsiedel. Network-wide Consensus Utilizing the Capture Effect in Low-power Wireless Networks. In *Proc. of SenSys*, 2017.

[17] B. A. Nahas, S. Duquennoy, and O. Landsiedel. Concurrent Transmissions for Multi-Hop Bluetooth 5. In *Proc. of EWSN*, 2019.

[18] Texas Instruments. CC2420 Datasheet.

[19] D. Vecchia, P. Corbalan, T. Istomin, and G. P. Picco. Playing with Fire: Exploring Concurrent Transmissions in Ultra-wideband Radios. In *in Proc. of SECON*, 2019.