Improving Broadcast Performance of Radio Duty-Cycled Internet-of-Things Devices

Salih Serdar Guclu, Tanir Ozcelebi, Johan J. Lukkien
Department of Mathematics and Computer Science
Eindhoven University of Technology, Eindhoven, The Netherlands
E-mail: {s.s.guclu, t.ozcelebi, j.j.lukkien}@tue.nl

***Abstract*—Asynchronous Radio Duty Cycling (ARDC) protocols can make embedded networked devices more energy efficient by keeping their radio off most of the time without a need for synchronization between devices. Some ARDC protocols can operate under 6LoWPAN adaptation layer in order to enable the vision of Internet-of-Things for battery operated devices. In this paper, we propose three different protocols which are modifications of the widely accepted ARDC protocol, ContikiMAC. The proposed solutions drastically improve energy efficiency and link layer delay for broadcast packets. Moreover, the proposed solutions are backward compatible with ContikiMAC and provide high reliability against frame reception errors. We present a detailed comparison with the legacy ContikiMAC and a standardized ARDC protocol, IEEE 802.15.4e Coordinated Sampled Listening (CSL), as well as the case of no duty cycling.**

I. Introduction

Low power and lossy networks (LLNs) are comprised of battery-operated and wirelessly networked resource constrained devices. These devices are building blocks of the Internet-of-Things (IoT) in many areas like sensing, logistics, surveillance and safety. However, resource constrained devices existed and were used for similar purposes before the emergence of IoT. Hence, these devices already had a standardized MAC and physical layer protocol, IEEE 802.15.4 [1], which enables low cost radios with low energy consumption. On the other hand, the IEEE 802.15.4 standard was not intended for the purpose of accessing these devices with the IPv6 protocol. 6LoWPAN (IPv6 over Low-power Wireless Personal Area Networks) [2] has emerged for making IEEE 802.15.4 compliant devices IPv6 accessible.

During the lifetime of an LLN node, idle listening constitutes the highest energy expense [3]. Battery operated devices avoid idle listening by turning-off their radio according to some scheme, which is called Radio Duty Cycling (RDC). As a subclass, Asynchronous RDC (ARDC) protocols operate without explicit synchronization between the nodes in order to evade the incurred overhead. ARDC protocols like ContikiMAC [4] and IEEE 802.15.4e Coordinated Sampled Listening (CSL) [5] natively support broadcasting, offer very low duty cycles and have flexible parameter settings for different application needs. The ContikiMAC protocol has been proven to fully support IPv6 through 6LoWPAN with years of battery lifetime [6].

IEEE 802.15.4 has two types of link layer transmissions: Unicasting and broadcasting. Unicasting is sending a MAC frame to a one-hop neighbor and expecting to receive an acknowledgement frame afterwards, while broadcasting is sending a MAC frame to all one-hop neighbors. In LLNs, broadcasting of MAC frames to one-hop neighbors is essential for multicasting protocols [7] and maintenance of routing protocols [8].

Broadcast MAC frames are not acknowledged in IEEE 802.15.4. Therefore, there is no way of informing the sender about delivery of the broadcast MAC frame. Although there were some efforts in the literature for ensuring the reliability of broadcast MAC frames by acknowledgements or negative-acknowledgements, they are not applicable for a general purpose IEEE 802.15.4 subnet. For example, [9] proposes an acknowledgement mechanism for IEEE 802.15.4 broadcasts, which requires to have knowledge of the topology. There were some other efforts for achieving reliable broadcasting by [10] and [11], but they operate under the assumption that broadcast MAC frames will be repeated by the receiver, which might not be the case. Moreover, [12] proposes a scalable broadcast acknowledgment mechanism for IEEE 802.11, by taking advantage of OFDM. However, their solution is not easily applicable to IEEE 802.15.4 subnets. To the best of our knowledge, there is no other work that is evaluating and enhancing the performance of broadcasting IPv6 packets on top of ARDC protocols.

The widely accepted ARDC protocol, ContikiMAC, implicitly relies on repetitions of broadcast MAC frames in order to achieve acceptable reliability. However, their approach increases link delay tremendously where sending a long IPv6 broadcast packet with default protocol parameters can take more than 1.4 seconds, which is huge compared to the 60 milliseconds maximum delay of the no duty cycling case. When the parameters of the ContikiMAC are adjusted in a more energy conservative manner, link delay of an IPv6 broadcast packet can be more than 5 seconds. Moreover, broadcasting a long IPv6 packet consumes 10 times more energy than sending the same packet with unicasting, while using the default parameters of ContikiMAC.

Although broadcasting in IEEE 802.15.4e CSL has small delay and low energy consumption, it does not provide any reliability precautions. When a receiver fails to receive a single broadcast MAC frame, the problem can only be solved in the application layer.

In this paper, we propose three different and backward-

978-1-5090-1328-9/16/$31.00 ©2016 IEEE

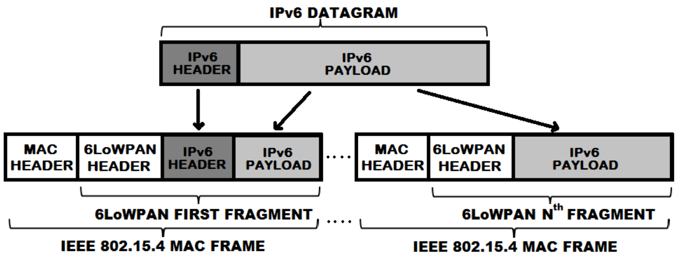


Fig. 1: IPv6 datagram divided into fragments by the 6LoWPAN adaptation layer and encapsulated in multiple IEEE 802.15.4 MAC frames.

compatible methods in order to achieve reliable, energy conserving and fast link layer transmissions for broadcasting IPv6 packets with duty cycling.

The paper is organized as follows: Section 2 gives a background about the existing protocols and standards. Section 3 introduces the methods we propose. Section 4 presents the comparative simulation results. The paper is concluded with section 5.

II. Background

A. 6LoWPAN

IEEE 802.15.4 is the de-facto physical layer and medium access layer standard for LLNs. However, it is not developed for accessing the wireless embedded devices with IP at all. The most fundamental problem that hampers the interaction between IEEE 802.15.4 and IPv6 is mismatch of Maximum Transmission Unit (MTU), i.e. IEEE 802.15.4 has an MTU of 127 bytes while IPv6 mandates an MTU of 1280 bytes. Moreover, IPv6 does not do fragmentation. Therefore, IEEE 802.15.4 cannot support IPv6 by default. In order to solve the problem, IETF proposed an adaptation layer in between these two layers: 6LoWPAN. Figure 1 shows the relation between MAC frames, 6LoWPAN fragments and the IPv6 datagram. Note that, unlike IPv4 fragmentation where the IP header is included in every fragment, 6LoWPAN sends the IPv6 header only in the first fragment.

The 6LoWPAN layer provides two related functionalities:

1) Header Compression: IPv6 header can occupy significant amount of space within an IEEE 802.15.4 MAC frame, leaving very small room for the IPv6 payload. 6LoWPAN addresses this issue with an optional header compression mechanism. IPv6 header compression is explained in [13]. Our methods can work with or without header compression mechanisms.

2) Fragmentation: 6LoWPAN layer divides an IPv6 datagram into multiple fragments such that each fragment can fit in an IEEE 802.15.4 MAC frame, i.e. the total size of the fragment plus the MAC header is less than or equal to the MTU of IEEE 802.15.4. The first 6LoWPAN fragment has the IPv6 header. Depending on the header length, a portion of the IPv6 payload can also be placed within the first fragment.

According to the protocol specification of 6LoWPAN, a receiver must expect to receive the fragments in any order. Therefore, a receiver might have to re-order the fragments. A receiver can infer whether it has received the entire set of fragments by checking the datagram size and datagram offset fields, which are part of the 6LoWPAN header. Upon reception of all fragments, the receiver first reorders them, and then uncompresses the IPv6 header if necessary. Subsequently, the IPv6 payload is extracted from the fragments and the fully merged datagram is sent to the IPv6 layer. In case the entire set of fragments are not received within a time-out period defined by 6LoWPAN, all received fragments are dropped. If the IPv6 datagram can fit into a single MAC frame, header fields that are related to fragmentation are not used.

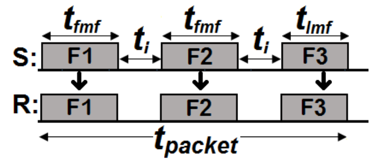


Fig. 2: Broadcasting without duty-cycling.

B. Broadcasting without duty-cycling (NO-RDC)

For comparison, we first introduce sending fragments without duty cycling, i.e. radio is always on. We name this case as NO-RDC. Receivers immediately capture the packets since their radio is always sampling the channel. Each fragment is encapsulated in an IEEE 802.15.4 MAC frame. If the sender detects collision or interference before starting to transmit the first fragment, the sender backs-off. In case the channel is free before the transmission of the first fragment, all the fragments are transmitted only once. Because broadcast MAC frames are never acknowledged, sender assumes that every neighbor has received every broadcast frame. However, this scheme fails to deliver the IPv6 packet to a neighbor when there is a frame reception error, since the neighbors are not able to notify the sender. We define *error resilience* as the receiver's ability to tolerate frame reception errors.

fc is the number of fragments for an IPv6 packet. Each one of the first $fc-1$ fragments occupies the maximum payload size of their corresponding MAC frame. The last fragment has a varying size since the IPv6 payload length may arbitrarily change. We define t_{fmf} as the duration of sending any of the first $fc-1$ fragments, including the physical layer and MAC layer headers. We also define t_{lmf} as the duration of sending the last fragment, including the physical layer and MAC layer headers. Between each consecutive MAC frame, there is a waiting time called t_i, which may change depending on the hardware. Figure 2 illustrates this mechanism.

Radio-on time of the sender while transmitting a single IPv6 datagram, average radio-on time of the receivers while receiving a single IPv6 datagram and the average link delay of receiving the IPv6 datagram are equal to each other:

$$t_{packet} = (t_{fmf} + t_i)(fc - 1) + t_{lmf} \quad (1)$$

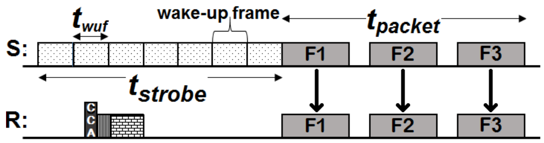


Fig. 3: Broadcasting in IEEE 802.15.4e CSL.

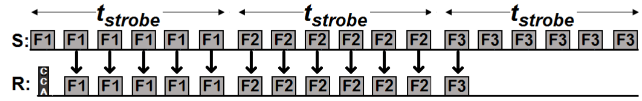


Fig. 4: Broadcasting in LEGACY.

C. 6LoWPAN compatible ARDC protocols

Not all ARDC protocols can work under the 6LoWPAN layer. The ARDC protocol must use IEEE 802.15.4 compliant frames. The time difference between receiving the first fragment and receiving the last fragment should not exceed the reception timeout of 6LoWPAN, regardless of the topology. Receiver initiated ARDC protocols fail to fully support 6LoWPAN since the typical way of broadcasting in receiver initiated ARDC protocols is sending multiple unicast frames to the one-hop neighbors [14]. Each fragment is encapsulated in a MAC frame, then sent as a unicast MAC frame to each neighbor. When the density of the network is high, long broadcast packets with multiple fragments are very likely to exceed the reception time-out of 6LoWPAN, preventing any node from receiving broadcast packets. A cross-layer approach can be used such that datagrams are sent in a round-robin fashion instead of sending the fragments in a round-robin fashion. In this case, reception times of the broadcast packets will vary significantly across receivers. Absence of broadcast support means that receiver initiated ARDC protocols are not suitable for a general purpose network stack. Therefore, this paper considers only the most advanced sender-initiated ARDC protocols, ContikiMAC and IEEE 802.15.4e CSL.

In sender initiated ARDC protocols, a receiver performs clear channel assessment (CCA) every t_{cycle} seconds. When a sender has a frame to send, it first wakes the receiver up. Waking the receiver up can be by repeatedly sending a special frame as in IEEE 802.15.4e CSL or by sending the actual frame repeatedly as in ContikiMAC.

After receiving the MAC frame, the receiver goes back to sleep until its next scheduled channel check time in both IEEE 802.15.4e CSL and ContikiMAC. While sending multiple fragments, this scheme increases the link delay with increasing fragment count and t_{cycle}. However, ARDC protocols utilize a special flag of the IEEE 802.15.4 MAC header for efficient transmission of subsequent fragments: Frame Pending Flag (FPF). If the FPF of a MAC frame is set, the receiver keeps its radio on for receiving another MAC frame. Correct usage of this flag can introduce better performance in terms of energy consumption and link delay.

1) IEEE 802.5.4e CSL: IEEE 802.5.4e CSL is a standardized variant of BoX-MAC-1 [15]. IEEE 802.5.4e CSL receivers measure the energy level of the channel every t_{cycle} seconds. The sender sends back-to-back wake-up frames for a duration of t_{strobe}, which is a duration slightly longer than t_{cycle} [16]. The purpose of this setting is ensuring the reception of the wake-up frame by every one-hop neighbor. Each wake-up frame contains a relative rendezvous time for the transmission of the MAC frame that contains the first fragment. After reception of a wake-up frame, receiver turns its radio off until the rendezvous time. Every wake-up frame has a fixed size, the time to transmit a wake-up frame is defined by t_{wuf}. Right after the back-to-back wake-up frame sequence, the first fragment is sent. Subsequent 6LoWPAN fragments are sent one after another, without an additional sequence of wake-up frames. In order to force the receiver to stay awake for the subsequent fragments, the FPF of every MAC frame is set except for the last one. Figure 3 illustrates this scheme.

IEEE 802.15.4e CSL sends every fragment only once. In addition, a receiver can fail to receive a wake-up frame. Therefore, IEEE 802.15.4e CSL is slightly less error resilient compared to NO-RDC, since IEEE 802.15.4e CSL receivers may also fail to receive the wake-up frame.

The average radio-on time of receivers is:

$$3t_{wuf}/2 + t_{packet} \tag{2}$$

The radio-on time of the sender and the average delay is:

$$t_{strobe} + t_{packet} \tag{3}$$

To the best of our knowledge, there is no publicly available implementation of IEEE 802.15.4e CSL. This is because the protocol requires to send back-to-back wake-up frames for long amounts of time while popular IEEE 802.15.4 compatible radios do not natively support this operation since staying in the TX mode for longer than the time to send the longest MAC frame is not mandated by IEEE 802.15.4.

2) Existing design of ContikiMAC (LEGACY): The ContikiMAC, the most advanced variant of the BoX-MAC-2 [15], is the default ARDC protocol of the Contiki OS protocol stack. Our proposed methods takes ContikiMAC as the baseline. ContikiMAC receivers measure the energy level of the channel every t_{cycle} seconds. ContikiMAC does not rely on wake-up frames. The broadcast MAC frame is repeatedly sent for a duration slightly longer than t_{cycle}, namely t_{strobe} [16].

While sending an IPv6 packet, the 6LoWPAN layer prepares the entire set of fragments and passes them to ContikiMAC. ContikiMAC puts each of those fragments into IEEE 802.15.4 compliant MAC frames. The FPF of every MAC frame is set except for the last one. Each MAC frame is repeatedly sent for a duration of t_{strobe}. A fragment is removed from the sending buffer after repeating its corresponding MAC frame for a duration of t_{strobe}.

This mechanism forces the receiver to be awake beginning from reception of the first fragment until reception of the last fragment, while receiving many duplicate MAC frames. Even though duplicates are dropped by ContikiMAC, receivers still spend energy and time for receiving duplicates, as depicted in Figure 4.

Average radio-on time of receivers is:

$$[(t_{strobe}+t_i+t_{fmf})/2]+t_i+(fc-2)(t_{strobe}+t_i)+t_{lmf} \quad (4)$$

Radio-on time of the sender can be calculated as follows:

$$(fc)(t_{strobe}+t_i)-t_i \quad (5)$$

Average delay becomes:

$$(fc-1)(t_{strobe}+t_i)+t_{lmf} \quad (6)$$

III. PROPOSED PROTOCOLS

The broadcasting mechanism of LEGACY creates significant link delay while transmitting broadcast packets. Moreover, it keeps the receiver's radio on for long durations, which drastically decreases the battery lifetime. However, LEGACY is advantageous in terms of error resilience thanks to multiple transmissions of the same MAC frame. Taking LEGACY protocol as the baseline, we propose three different protocols for broadcasting IPv6 datagrams. Our solutions are backward compatible for receivers of LEGACY and they improve the link delay, the energy consumption and the bandwidth usage.

A. SLEEPY

This approach improves energy consumption of receivers and slightly increases the delay compared to LEGACY. Sender utilizes LEGACY with a simple modification: Never set the FPF for any of the broadcast MAC frames. This is because a broadcast strobe is long enough to guarantee reception of the MAC frame of that fragment by every neighbor. The SLEEPY sender employs the same sending buffer management policy with LEGACY.

No changes are needed in receivers and they can utilize LEGACY. Receivers go back to sleep after reception of each fragment since FPF is not set by the sender. Next channel check of the receiver will fall into the next fragment's broadcast strobe. After receiving that fragment, receivers will go back to sleep again. This scheme is significantly more energy efficient for receivers, compared to LEGACY. SLEEPY also provides error resilience since receivers still have the chance of trying to receive the MAC frames that they have failed to receive. This is because the receiver checks the channel in halfway through the t_{strobe} in the average case, which gives the ability of trying to receive the failed MAC frame again.

Average radio-on time of receivers is:

$$3t_{packet}/2-(fc-1)t_i/2 \quad (7)$$

Radio-on time of the sender can be calculated as follows:

$$fc(t_{strobe}+t_i)-t_i \quad (8)$$

Average delay becomes:

$$(fc-1)(t_{strobe}+t_i)+t_{lmf}+(t_{strobe}+t_i)/2 \quad (9)$$

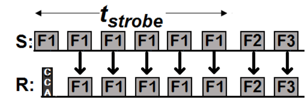


Fig. 5: Broadcasting in SYNCED.

B. SYNCED

SYNCED follows a similar method as IEEE 802.15.4e CSL. No changes are needed in receivers and they can utilize LEGACY. The sender repeats the MAC frame of the first fragment like a regular broadcasting MAC frame throughout t_{strobe}, such that every receiver receives it. Remaining fragments are sent only once. The sender sets the FPF of every MAC frame except for the last one. Thus, all the neighbors start to receive the subsequent fragments at the same time.

Duplicate receptions of the MAC frame of the first fragment are discarded by the MAC layer, as depicted in Figure 5. The first fragment is removed from the sending buffer after repeating its MAC frame for a duration of t_{strobe}. Each subsequent fragment is removed after transmission of its MAC frame.

Average radio-on time of receivers is:

$$(t_{strobe}+t_i-t_{fmf})/2+t_{packet} \quad (10)$$

Radio-on time of the sender and average delay becomes:

$$t_{strobe}+t_{packet}-t_{fmf} \quad (11)$$

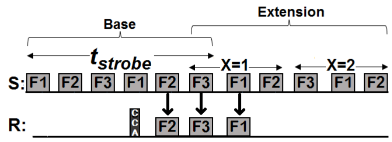
C. X-CIRCULAR

X-CIRCULAR is a cross layer approach between 6LoWPAN and ContikiMAC, repeatedly sending the fragments in a circular manner.

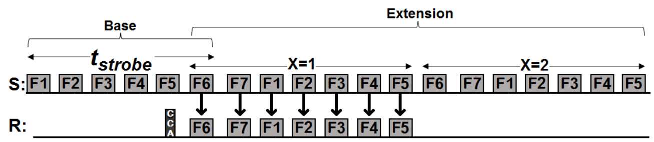
X-CIRCULAR sends the fragments in two subsequent steps, the base step and the extension step. In base, MAC frames of the fragments are repeated in a circular manner for a duration of t_{strobe}. During extension, MAC frames of all fragments are repeated X times where X>0. The last fragment of the base is counted as the first fragment of extension. Figure 6 illustrates X-CIRCULAR for X=2. Higher values of X increases the error resilience at the cost of increasing bandwidth usage and energy consumption of the sender. Changing the X value has no effect on the receiver's reception mechanism or energy consumption or average link delay i.e. time to receive the entire set of fragments. Therefore, senders are free to change the X value for different IPv6 packets, without even notifying the receivers.

In X-CIRCULAR, FPFs of the MAC frames must always be set in order not to allow any of the receivers to terminate reception by sleeping. The sending buffer is cleared after the transmission of the entire fragment set.

Setting every MAC frame's FPF causes receivers to receive duplicate frames, even after they have already received the entire IPv6 datagram. Moreover, receivers expect to receive more frames even after the sender has stopped sending since the last MAC frame had indicated that there will be more frames by setting the FPF. The receiving side of X-CIRCULAR is designed for preventing duplicate reception of MAC frames.



(a) Fragment count is low, i.e. $t_{packet} < t_{strobe}$.



(b) Fragment count is high, i.e. $t_{packet} > t_{strobe}$.

Fig. 6: Broadcasting in X-CIRCULAR for X=2.

LEGACY receiver's MAC frame sequence number buffering mechanism stores only one entry for each sender, which causes the receiver to forget the ones preceding than the last MAC frame. When we use a LEGACY receiver with an X-CIRCULAR sender, the IPv6 datagram is successfully received and merged first, but upcoming duplicate frames are passed to the 6LoWPAN layer due to failing duplicate check mechanism of LEGACY. This creates severe problems since the 6LoWPAN layer does not check duplicate fragments.

We first modified the MAC frame sequence number buffering mechanism as a first-in first-out queue (FIFO), allowing multiple entries for a sender. This modification does not affect other operations of the protocol stack and sufficient for problem-free reception of IPv6 datagrams since all the duplicates are suppressed in the MAC layer. However, receiving the duplicates is still to be avoided for conserving energy.

After completely receiving the IPv6 datagram, 6LoWPAN layer of the X-CIRCULAR receiver signals the ContikiMAC layer for turning the radio off. However, this mechanism itself is not sufficient for avoiding to receive duplicates. When the X value is high or the datagram size is long, some of the receivers will do their next channel check while the sender is still repeating the fragments. They will wake up again and start receiving duplicates.

Therefore, an X-CIRCULAR receiver employs the following mechanism: Upon reception of the entire fragment set, the 6LoWPAN layer signals the ContikiMAC for turning the radio off. If ContikiMAC is awaken by a duplicate broadcast MAC frame in its next channel check, the radio is immediately turned-off until the next channel check, i.e. FPF of the received MAC frame is ignored. This behavior continues until receiving a MAC frame that is not duplicate.

A duplicate frame before the 6LoWPAN's signal to ContikiMAC can only occur when some fragments are missed. In this case, the duplicate is just discarded and the radio is kept on for receiving the missing fragment. This strategy allows the receiver to discard duplicates until the reception of the missing fragment(s).

Average radio-on time of receivers is:

$$t_{fmf}/2 + t_i + t_{packet} \tag{12}$$

Radio-on time of the sender can be calculated as follows:

$$t_{strobe} + X(t_{packet} + t_i) - t_{fmf} \tag{13}$$

Average delay becomes:

$$(t_{strobe} + t_i)/2 + t_{packet} \tag{14}$$

IV. Simulation Results & Discussions

We have tested our methods in the COOJA simulator [17], using Wismote with CC2520 radio transceiver. We run the simulations for the following channel check rates (i.e. $1/t_{cycle}$): 2, 8, 16 and 64 Hertz. Figure 7 depicts average radio-on time of receivers, which has a nearly-linear relation with the receiver's energy consumption while receiving a single IPv6 datagram. Figure 8 presents the radio-on time of the sender, which also determines the bandwidth usage and sender's energy consumption while transmitting an IPv6 datagram. Figure 9 illustrates the average link delay of receiving an IPv6 datagram. Figure 10 shows the probability of successfully receiving the IPv6 datagram where each receiver fails to receive a MAC frame with 10% or 30% probability.

Simulations show that link delay of SYNCED is similar to IEEE 802.15.4e CSL and much lower than LEGACY and SLEEPY. In terms of energy efficiency, radio-on time of SYNCED receivers is much less than LEGACY. However, SYNCED is not resilient to errors like IEEE 802.15.4e CSL.

In dense networks, reducing the energy consumption of receiver is more important than reducing the energy consumption of the sender while broadcasting. Therefore, SLEEPY effectively improves energy efficiency in dense networks. However, SLEEPY cannot decrease the link delay of LEGACY.

According to the simulation results, datagrams with a higher number of fragments have a lower chance of successful reception. However, X-CIRCULAR can enhance the probability of successful reception by freely increasing X value for long packets. Moreover, if dynamically changing error rates of the channel are known, X can be adaptively set. In many realistic scenarios, X-CIRCULAR achieves very high probability of successful transmission. In addition, X-CIRCULAR provides the lowest energy consumption for receivers and the lowest average link delay, which is very close to NO-RDC.

In terms of backward compatibility, LEGACY receivers can fully benefit from the advantages of SLEEPY and SYNCED, but not X-CIRCULAR's. However, replacing the MAC frame sequence number buffering mechanism with FIFO queue is sufficient for being able to receive the IPv6 datagram from the X-CIRCULAR sender. This minor modification does not hamper the other operations of the protocol stack and it allows LEGACY receiver to have a short delay and error resilience just like an X-CIRCULAR receiver. In this case, energy consumption of the LEGACY receiver will be higher than the X-CIRCULAR receiver since the LEGACY receiver will not turn its radio off until the end of the MAC frame strobe. In spite of this, the radio-on time of the LEGACY receiver while listening to an X-CIRCULAR sender is still much smaller than the radio-on time of the same receiver while listening

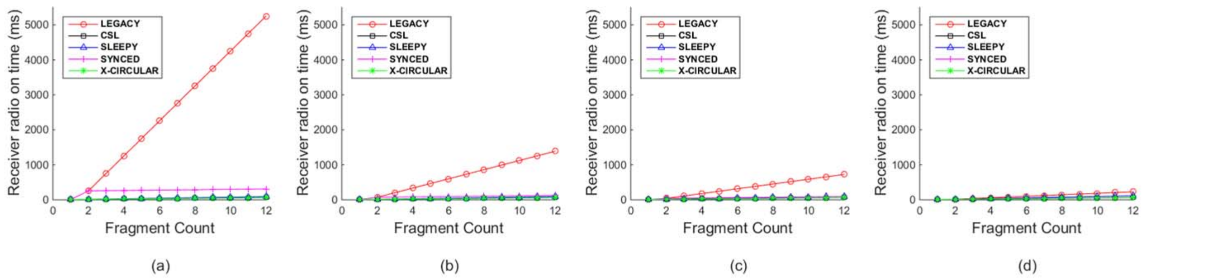


Fig. 7: Receiver's average radio-on time for various channel check rate (CCR) values: a)2, b)8, c)16 and d)64 Hz.

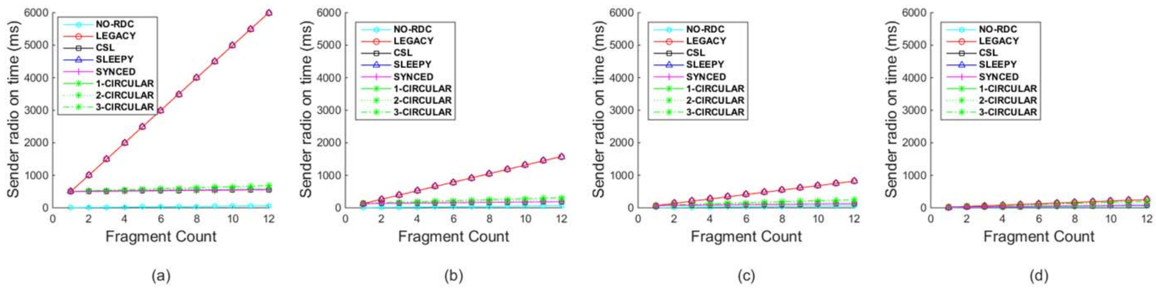


Fig. 8: Sender's radio-on time for various CCR values: a)2, b)8, c)16 and d)64 Hz.

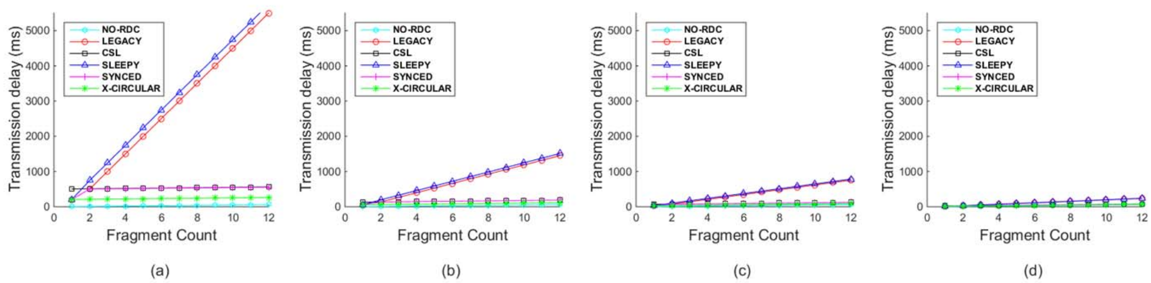


Fig. 9: Average link delay for various CCR values :a)2, b)8, c)16 and d)64 Hz.

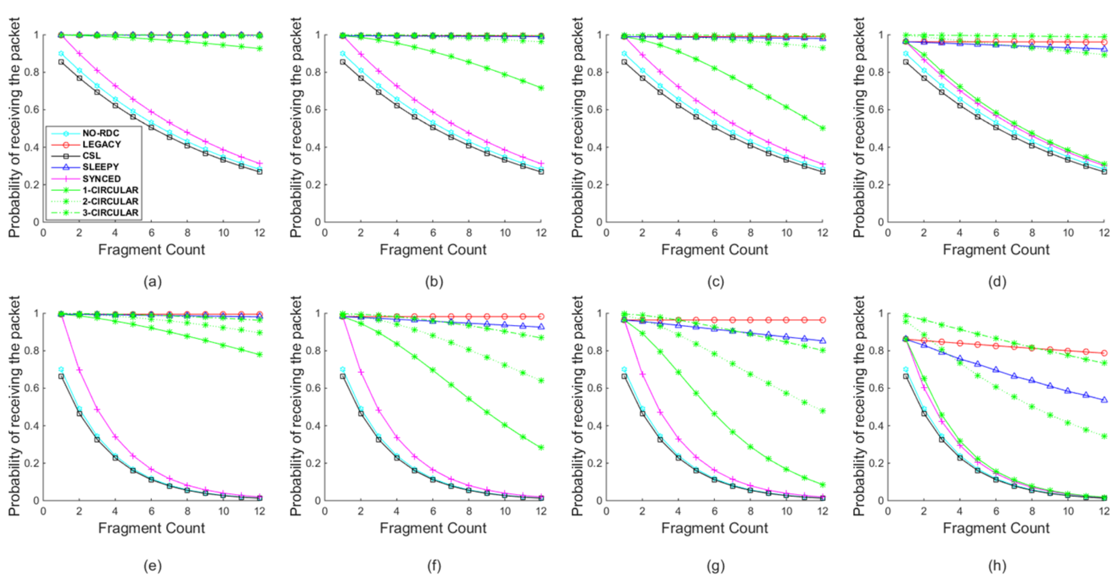


Fig. 10: Probability of successful datagram reception for various CCR values: a)2, b)8, c)16 d)64, e)2, f)8, g)16 and h)64 Hz. Probability of MAC frame reception failure is 10% for subgraphs a-d and 30% for subgraphs e-h.

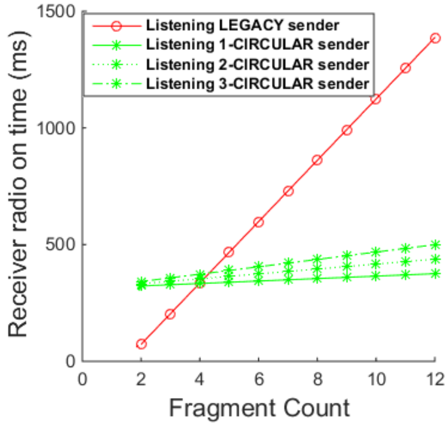


Fig. 11: Comparison of LEGACY receiver's average radio-on time for various senders. CCR is set to its default value, which is 8 Hz. LEGACY receiver's MAC frame sequence number buffering mechanism is replaced with FIFO.

TABLE I: Interoperability Matrix of the Protocols

		Receiver				
		NO-RDC	LEGACY	SLEEPY	SYNCED	X-CIRC.
Sender	NO-RDC	Yes	No	No	No	No
	LEGACY	Yes	Yes	Yes	Yes	Yes
	SLEEPY	Yes	Yes	Yes	Yes	Yes
	SYNCED	Yes	Yes	Yes	Yes	Yes
	X-CIRC.	Yes with FIFO	Yes with FIFO	Yes with FIFO	Yes with FIFO	Yes

TABLE II: Comparison of Broadcasting Protocols

Method	Delay	Radio on time (receiver)	Radio on time (sender)	Error resilience	Backward compatibility with LEGACY
NO-RDC	Low	N/A	N/A	None	See Table I
CSL	Low	Low	Low	None	No
LEGACY	High	High	High	High	Yes
SLEEPY	High	Low	Low	High	Yes
SYNCED	Low	Low	Low	None	Yes
X-CIRCULAR	Low	Low	Low	High	Partially

to a LEGACY sender for most scenarios. Figure 11 depicts the comparison of LEGACY receiver's average radio-on time between 1, 2 and 3-CIRCULAR senders and LEGACY sender.

Table I presents the interoperability relations of the protocols. Table II presents a comparison of the protocols according to the simulation results.

V. Conclusions

This paper proposes three different protocols for efficient broadcasting of IPv6 packets in ARDC protocols. We first presented the problems of the ContikiMAC and IEEE 802.15.4e CSL. The broadcasting method of ContikiMAC has high delay and high energy consumption with high reliability. On the other hand, IEEE 802.15.4e CSL has very low delay and low energy consumption but it lacks reliability. We introduced the SLEEPY protocol, which significantly improves the energy consumption of the ContikiMAC and provides backward compatibility. Then, we introduced another backward compatible protocol for Contiki OS protocol stack, SYNCED, which is similar to IEEE 802.15.4e CSL. Finally, we introduced the X-CIRCULAR protocol, which provides very small link delay, very low energy consumption, acceptable backward compatibility and high reliability.

VI. Acknowledgements

The authors would like to thank Dr. Richard Verhoeven for his valuable comments. This research is enabled by the OpenAIS project, supported by the Horizon 2020 program of the European Union under grant agreement number 644332.

References

[1] "Ieee standard for local and metropolitan area networks part 15.4: Low-rate wireless personal area networks (lr-wpans)," IEEE 802.15.4, 2011.
[2] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks," RFC4944, IETF, 2007.
[3] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-mac: A short preamble mac protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, ser. SenSys '06. ACM, 2006, pp. 307–320.
[4] A. Dunkels, "The contikimac radio duty cycling protocol," SICS, Tech. Rep. T2011:13, Dec. 2011.
[5] "Ieee standard for local and metropolitan area networks part 15.4: Low-rate wireless personal area networks (lr-wpans) amendment 1: Mac sublayer," IEEE 802.15.4e, 2012.
[6] A. Dunkels, J. Eriksson, and N. Tsiftes, "Low-power interoperability for the ipv6-based internet of things," in *Proceedings of the 10th Scandinavian Workshop on Wireless Ad-hoc Networks (ADHOC 11)*.
[7] J. Hui and R. Kelsey, "Multicast Protocol for Low-Power and Lossy Networks (MPL)," RFC7731, IETF, 2016.
[8] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC6550, IETF.
[9] T. wen Sung, T. ting Wu, C. sing Yang, and Y. min Huang, "Reliable data broadcast for zigbee wireless sensor networks," in *International journal on smart sensing and intelligent systems*, 2010, pp. 504–520.
[10] J. Thriveni, B. Ashwini, A. Latha, K. R. Sandhyashree, G. L. Prakash, K. R. Venugopal, and L. M. Patnaik, "Broadcast with negative acknowledgements for enhanced delivery ratio in manet with dual coverage," in *Wireless Days, 2008. WD '08. 1st IFIP*, Nov 2008, pp. 1–5.
[11] R. Liu, T. Luo, L. Zhang, J. Li, and S. Fang, "A forwarding acknowledgement based multi-hop broadcast algorithm in vehicular ad hoc networks," in *Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on*, Dec 2012, pp. 1258–1262.
[12] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "Smack: A smart acknowledgment scheme for broadcast messages in wireless networks," *SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 4, pp. 15–26, 2009.
[13] J. Hui and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks," RFC6282, IETF, 2011.
[14] X. Fafoutis, A. D. Mauro, M. D. Vithanage, and N. Dragoni, "Receiver-initiated medium access control protocols for wireless sensor networks," *Computer Networks*, vol. 76, pp. 55 – 74, 2015.
[15] D. Moss and P. Levis, "Box-macs: Exploiting physical and link layer boundaries in low-power networking," *Computer Systems Laboratory Stanford University*, pp. 116–119, 2008.
[16] S. S. Guclu, T. Ozcelebi, and J. J. Lukkien, "Dependability analysis of asynchronous radio duty cycling protocols," in *25th International Conference on Computer Communication and Networks (ICCCN)*, 2016.
[17] F. Österlind, "A sensor network simulator for the Contiki OS," *SICS Research Report*, 2006.