Poster Abstract:

The Low-Power Wireless Bus: Simplicity is (Again) the Soul of Efficiency

Federico Ferrari* Marco Zimmerling* Lothar Thiele* Luca Mottola†
*Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland
†Swedish Institute of Computer Science (SICS), Kista, Sweden
{ferrari, zimmerling, thiele}@tik.ee.ethz.ch luca@sics.se

ABSTRACT

We present the *low-power wireless bus* (LWB), a simple yet efficient communication support for low-power wireless networks. The LWB maps different communication demands onto fast Glossy network flooding, effectively turning the wireless network into a bus-like infrastructure. The LWB requires no information of the network topology, thus drastically reducing the control overhead of common solutions such as route maintenance, and natively supports many-to-many communication and mobile nodes in addition to more traditional static, one-to-many scenarios. For instance, experiments on a 90-node testbed show that on average the LWB reduces packet loss by a factor of 231 and energy consumption due to communication by a factor of 11 compared to a state-of-the-art many-to-many routing protocol.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*wireless communication*

Keywords

Wireless Sensor Networks, Multi-Sink, Flooding, Shared Bus

1. OVERVIEW

Common communication support for low-power wireless networks mostly battled one of the defining features of the wireless channel: that of being a broadcast medium. Existing network-layer protocols build and maintain routes connecting sources to destinations, in an attempt to spare efforts from nodes that may be excluded from processing. The routes tend to be tailored to specific traffic patterns (*e.g.*, one-to-many or many-to-many) and application scenarios (*e.g.*, presence or absence of mobile nodes). However, route maintenance may require significant control traffic against link fluctuations and topology changes. Moreover, existing solutions typically require multiple functionality at different layers, whose interactions may be difficult to understand [1].

A different take on the problem. Owing to the broadcast nature of the wireless channel, the LWB incarnates a different approach, intuitively described in Fig. 1. It maps all communication requests onto network flooding by using Glossy, which provides 99.99 % flooding reliability with millisecond latencies in networks of hundreds of nodes, requiring no information of the network topology [2]. This effectively turns a multi-hop wireless network into a bus-like infrastructure. As a result, building and maintaining distributed routing state becomes unnecessary. Glossy's global time synchronization allows to integrate radio duty-cycling into the LWB, sparing the need for multiple network layers.

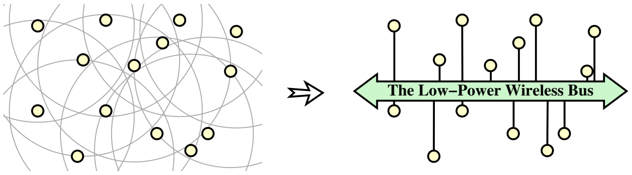


Figure 1: The LWB provides connectivity in multi-hop wireless networks similar to a shared bus.

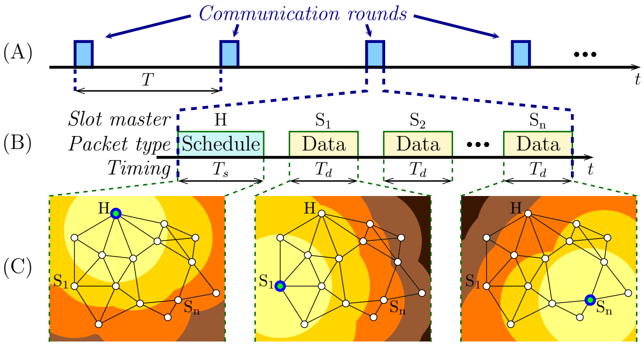
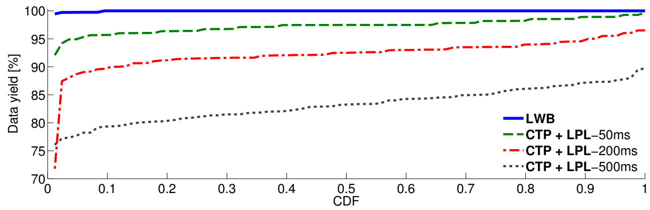


Figure 2: Communication rounds (A); communication slots in a round (B); Glossy floods in a slot (C).

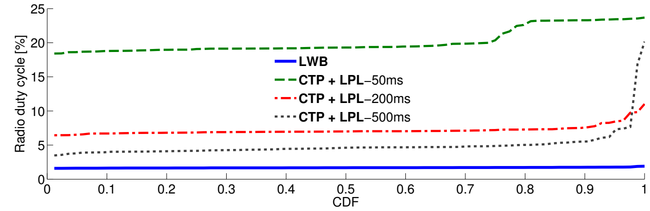
The absence of network state allows the LWB to (*i*) natively support multiple communication patterns with the *same* implementation, and (*ii*) be inherently resilient to topology changes. For instance, no changes to the protocol operation are required to support communication to multiple sinks, typical of control scenarios [9], or in presence of mobile nodes. The integrated design and simplicity of the LWB's functioning also make it straightforward to set the (few) protocol parameters and to understand its operation.

The LWB in a nutshell. Nodes in the LWB communicate in a *time-triggered* fashion [6] based on Glossy's built-in time synchronization. At most one *master* node at a time puts a message on the bus (*i.e.*, initiates a Glossy flood), whereas the other *slave* nodes read the message from the bus (*i.e.*, receive and relay the flooding packet). Communication complies with a *global schedule* that rotates the master role over time among all source nodes. A dedicated *host* node periodically announces the schedule. Depending on the application scenario, the schedule can be static or dynamically

IPSN'12, April 16–20, 2012, Beijing, China.
ACM 978-1-4503-1227-1/12/04.



(a) Data yield.



(b) Radio duty cycle.

Figure 3: CDFs showing the performance of the LWB against CTP + LPL on Twist with a single sink.

re-computed, for example, based on bandwidth demands. We are currently investigating several scheduling policies.

In our current prototype, communication over the LWB occurs within *communication rounds* repeated with period T, as shown in Fig. 2(A). Nodes keep their radio off between two rounds to save energy. At the beginning of a round, the host node H transmits the current schedule, as illustrated in Fig. 2(B). This initial Glossy flood also time-synchronizes the nodes. The rest of the round is divided into non-overlapping *communication slots* wherein every source node S_i is granted access to the bus. Each communication slot then corresponds to a distinct Glossy flood, as shown in Fig. 2(C).

To ensure exclusive bus access by the master node against communication failures, a node is permitted to initiate a flood *only* if it received the last schedule from the host, which also provides time synchronization.

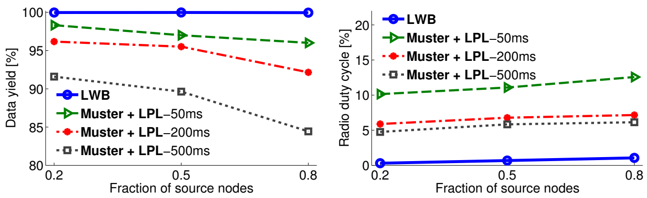
2. PRELIMINARY RESULTS

Our preliminary results from several diverse scenarios confirm that *"simplicity is the soul of efficiency"* [3]; that is, the cost of acquiring and maintaining distributed routing state often outweighs the benefits it possibly brings.

Experimental settings. We consider periodic data collection applications and corresponding performance metrics [4]: (*i*) *data yield* as the fraction of packets successfully received at the sink over those sent; and (*ii*) *radio duty cycle* as the fraction of time a node keeps the radio on, which provides a measure of a protocol's energy efficiency.

We compare the current LWB prototype with state-of-the-art routing and MAC protocols in both many-to-one and many-to-many scenarios. We use the Collection Tree Protocol (CTP) [4] available in TinyOS 2.1.1 as a baseline for the former and Muster [7] for the many-to-many case. Both protocols run atop the low-power listening (LPL) layer [8]. We repeat the experiments with 50 ms, 200 ms, and 500 ms LPL wake-up intervals. In all experiments, the application payload is 15 bytes. In the LWB, the round period T is set to 1 min, equal to the inter-packet interval. All operational parameters of CTP and Muster are at their default values.

The experiments run on the Twist [5] testbed, a 90 TelosB nodes installation densely spread across three floors in a university building. We set the transmit power to -7 dBm, yielding a maximum network diameter of 3 hops, and use channel 26 to limit the interference with co-located WiFi. We employ Contiki's power profiler to measure the radio duty cycle of the LWB prototype in software, and adopt a similar approach in TinyOS for all other protocols.



(a) Data yield. (b) Radio duty cycle.

Figure 4: LWB against Muster + LPL on Twist, with varying number of source nodes and 8 sinks.

Results. Fig. 3 shows CDFs comparing the performance of the LWB prototype against CTP + LPL in a setting where all nodes funnel data to a single sink. The LWB outperforms CTP + LPL for all wake-up intervals we tested. It achieves a data yield of 100 % for most nodes (see Fig. 3(a)) with an average of 99.97 %, and performs evenly in radio duty cycle with an average of 1.69 % (see Fig. 3(b)). In comparison, CTP trades data yield for radio duty cycle, achieving the highest average data yield of 97.31 % at 50 ms wake-up interval, but at 20.18 % average radio duty cycle. Longer wake-up intervals result in significantly lower data yield, averaging 83.19 % at 500 ms wake-up interval.

Fig. 4 depicts the LWB performance against Muster + LPL in a setting with 8 sinks and a varying number of source nodes. The *unmodified* LWB used for the single-sink experiments outperforms a protocol conceived for multi-sink scenarios. The average data yield across all sinks with the LWB is steadily at 99.97 %, whereas with Muster it starts at 98.32 % and decreases to 96.01 % with more sources due to route maintenance overhead (see Fig. 4(a)). Moreover, the LWB achieves between 0.31 % and 1.07 % radio duty cycles, whereas Muster's highest data yield corresponds to an average duty cycle between 10.14 % and 12.57 % (see Fig. 4(b)).

Acknowledgments. This work was supported by Nano-Tera, the National Competence Center in Research on Mobile Information and Communication Systems under SNSF grant number 5005-67322, the Swedish Foundation for Strategic Research, and the Cooperating Objects Network of Excellence under contract number EU-FP7-2007-2-224053.

3. REFERENCES

[1] J. Choi, M. Kazandjieva, M. Jain, and P. Levis. The case for a network protocol isolation layer. In *ACM SenSys*, 2009.

[2] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. Efficient network flooding and time synchronization with Glossy. In *ACM/IEEE IPSN*, 2011.

[3] A. Freeman. *The eye of Osiris.* Cambridge University Press, 1996.

[4] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. Collection tree protocol. In *ACM SenSys*, 2009.

[5] V. Handziski, A. Köpke, A. Willig, and A. Wolisz. TWIST: A scalable and reconfigurable testbed for wireless indoor experiments with sensor networks. In *ACM REALMAN*, 2006.

[6] H. Kopetz. The time-triggered model of computation. In *IEEE RTSS*, 1998.

[7] L. Mottola and G. P. Picco. Muster: Adaptive energy-aware multi-sink routing in wireless sensor networks. *IEEE Trans. on Mobile Computing*, 10(12), 2011.

[8] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *ACM SenSys*, 2004.

[9] J. A. Stankovic, I. Lee, A. Mok, and R. Rajkumar. Opportunities and obligations for physical computing systems. *Computer*, 38, 2005.