

The Bus goes Wireless: Routing-Free Data Collection with QoS Guarantees in Sensor Networks

Federico Ferrari*

Marco Zimmerling*

Lothar Thiele*

Luca Mottola†

*Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland

†Swedish Institute of Computer Science, Kista, Sweden

{ferrari, zimmerling, thiele}@tik.ee.ethz.ch

luca@sics.se

Abstract—We present the *low-power wireless bus (LWB)*, a new communication paradigm for QoS-aware data collection in low-power sensor networks. The LWB maps all communication onto network floods by using Glossy, an efficient flooding architecture for wireless sensor networks. Therefore, unlike current solutions, the LWB requires no information of the network topology, and inherently supports networks with mobile nodes and multiple data sinks. A LWB prototype implemented in Contiki guarantees bounded end-to-end communication delay and duplicate-free, in-order packet delivery—key QoS requirements in many control and mission-critical applications. Experiments on two testbeds demonstrate that the LWB prototype outperforms state-of-the-art data collection and link layer protocols, in terms of reliability and energy efficiency. For instance, we measure an average radio duty cycle of 1.69 % and an overall data yield of 99.97 % in a typical data collection scenario with 85 sensor nodes on Twist.

I. INTRODUCTION

Providing and analyzing Quality of Service (QoS) guarantees are key factors in making low-power wireless networks viable for a variety of applications, including safety-critical alarm systems, real-time control for factory automation, and mission-critical patient monitoring. These applications rely on a data collection service that provides bounded end-to-end delays, highly reliable packet delivery, and energy efficiency to achieve system lifetimes of several years. However, current solutions often fall short of these requirements, especially in multi-hop networks involving mobile nodes and multiple sinks.

A major obstacle in current solutions is their strong dependency on the network topology. To cope with the multi-hop nature of low-power wireless networks, they let nodes continuously collect information about link qualities and neighboring nodes, serving, for example, as input to routing and TDMA scheduling algorithms [11], [15]. Keeping such network state up-to-date in a wireless environment, where channel conditions change frequently [27], consumes significant bandwidth and energy. Node mobility compounds the problem as network state becomes quickly outdated [9]. Moreover, many existing solutions use multiple protocols concurrently, which can lead to unintended interactions that impair system performance and cause failures difficult to find and fix [8], [25], or they employ cross-layer designs, which often increase complexity and thus hamper analysis of the provided QoS guarantees [26].

To tackle the issues above, we adopt a clean-slate design and rethink the entire networking stack. In particular, we propose the *low-power wireless bus (LWB)*, a novel communication paradigm for data collection with QoS guarantees.

As described in Sec. II, the LWB achieves virtual single-hop connectivity in multi-hop wireless networks by mapping all communication onto Glossy floods [12]. Glossy provides network flooding and global time synchronization, achieving a reliability higher than 99.99 % and latencies of a few milliseconds in networks of about 100 nodes, without requiring any knowledge of the current network topology [12]. Using the LWB, a *host* node first floods a packet containing a *global communication schedule*. After receiving this packet, all nodes are *time-synchronized* and communicate in a *time-triggered* fashion [19] according to the schedule. Subsequent floods by the host serve to keep the nodes synchronized and update the global communication schedule, if necessary. In this way, the LWB connects nodes in a low-power wireless network much like a wired bus connects components inside a computer.

As a result of our design, the LWB requires no information about the network topology—a synchronized time and a global communication schedule fully determine its state. The LWB thus provides a routing-free data collection service with inherent support for mobile nodes and multiple sinks, while disposing of link layers, link estimators, and neighbor tables. Moreover, a prototype implementation of the LWB, described in Sec. III, provides three important QoS guarantees: bounded end-to-end delay and duplicate-free, in-order packet delivery.

At the same time, the LWB prototype is also highly reliable and energy-efficient, as demonstrated by experiments on Twist and a local testbed. For instance, we find in Sec. IV that the LWB provides an average data yield above 99.45 % across a variety of scenarios, ranging from low data rate to high data rate, from static nodes to mobile nodes, and under wireless interference. On 85 nodes on Twist, the LWB achieves an average radio duty cycle of 1.69 %, while delivering 99.97 % of all packets generated with an inter-packet interval of 1 min by all nodes. We also compare the performance of the LWB with CTP on top of TinyOS low-power listening, and find that it outperforms the latter consistently in all scenarios we tested.

II. THE LOW-POWER WIRELESS BUS

This section describes the main concepts and implications of the *low-power wireless bus (LWB)*. We first motivate the use of network flooding for data collection by addressing the following question: *Can network flooding serve as a viable basis for communication in low-power wireless networks?*

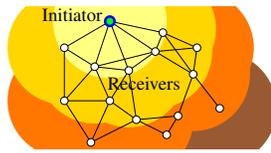


Figure 1. Example of packet propagation during a Glossy network flood. Nodes within the same color area relay the same packet at the same time.

A. Flooding for Communication

Network flooding naturally supports one-to-many communication needed for several important tasks in real-world applications [7]. Such tasks include network reprogramming to change or add functionality in a deployed system, runtime adjustment of configuration parameters, and dissemination of maintenance commands. Many-to-one communication is instead characteristic of data collection systems, where a set of source nodes reports sensor readings to a sink node, typically along the multi-hop paths of a routing topology [13]. Alternatively, each source node could simply initiate different network floods, and the sink would be among the recipients.

Network flooding, however, has a reputation of being costly, especially in low-power wireless networks with their severe bandwidth and energy constraints. So far, the use of flooding for communication has been considered prohibitively expensive, and thus impractical [2]. Glossy, a flooding architecture for wireless sensor networks [12], provides unique properties in terms of efficiency and reliability that fundamentally change this perception, paving the way for our main contribution.

Using Glossy, one node (*initiator*) floods a packet to all other nodes (*receivers*) in the network. As a node receives the packet, it can also accurately time-synchronize with the initiator. More specifically, Glossy leverages *synchronous transmissions*: nodes receiving the flooding packet relay it at the same time. Fig. 1 exemplifies how a packet propagates through a multi-hop network during a Glossy flood. Nodes within the same color area receive the packet at the same time; as soon as their receptions end, they immediately trigger a transmission and synchronously relay the same packet. Glossy makes the temporal offset among these synchronous transmissions not exceed $0.5 \mu\text{s}$ with very high probability. In this way, the baseband signals (modulated IEEE 802.15.4 symbols) *interfere constructively* and receivers can correctly decode the packet.

In testbeds of about 100 nodes, Glossy achieves flooding latencies of a few milliseconds; nodes typically receive the flooding packet with a probability above 99.99%, and have their radios turned on for only a few milliseconds during a flood. Moreover, Glossy provides accurate global time synchronization with an average error below $1 \mu\text{s}$ [12]. Most importantly, Glossy requires *no knowledge of the network topology*: all nodes relay a packet as soon as they receive it, independent of their current position and neighboring nodes.

B. Our Approach

The LWB uses *exclusively* Glossy network floods for communication. Glossy provides one-to-many communication in multi-hop wireless networks without requiring any knowledge

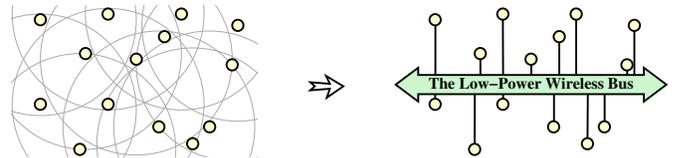


Figure 2. Using only Glossy network floods for communication, the LWB provides virtual single-hop connectivity in multi-hop wireless networks.

of their topology. As a result, the LWB hides the multi-hop nature of a network and achieves *virtual single-hop connectivity* among all nodes. Therefore, nodes communicate with each other as if they were connected through a shared bus, as shown in Fig. 2. At any point in time, at most one node (*master*) initiates a flood, while the remaining nodes (*slaves*) receive and relay the packet. That is, the master puts a message on the bus, and the slaves read the message from the bus.

To ensure that different network floods never overlap, the LWB exploits Glossy's global time synchronization: all floods are confined within sequential *communication slots* and *only time-synchronized* nodes participate in the communication. Nodes communicate in a *time-triggered* fashion [19] according to a *global schedule* that distributes the master role over time. A dedicated *host* node announces the schedule. Depending on the concrete LWB implementation and the application scenario, the schedule can be static or dynamically re-computed, for example, based on bandwidth requests issued by nodes.

C. Implications

We now detail the implications of virtual single-hop connectivity on the design of low-power wireless systems.

Routing-free communication. Data collection protocols usually require nodes to *explicitly* determine how a packet propagates through the network. That is, they force nodes to continuously identify their neighbors and evaluate the quality of links according to some metric [13]. Keeping such network state information up-to-date often introduces significant energy overhead, especially when the wireless channel conditions change frequently (*e.g.*, due to interference or mobility). Moreover, properties such as consistent neighborhood views among nodes [17] or loop-free routing [13] are not trivial to fulfill. Thus, low-power wireless systems typically use best-effort approaches and often fail to provide QoS guarantees, such as bounded end-to-end delay or in-order, duplicate-free packet delivery. Users may need to employ offline algorithms to reconstruct the correct temporal order of received data [18].

By contrast, Glossy network floods do not require nodes to explicitly route packets or maintain any knowledge of the network topology. In fact, a synchronized time and a global communication schedule suffice, and fully determine the state of the LWB. As a result, the state space of the entire system is drastically reduced, simplifying the analysis of important system properties. We highlight in Sec. III the QoS guarantees provided by a prototype implementation of the LWB.

Not yet another collision-free TDMA. The LWB avoids by design collisions between different packets: it guarantees that at a certain time there is always at most one master node, that is, at most one initiator of a flood. To this end, a

global schedule—the same for the whole network—assigns sequential, non-overlapping communication slots to nodes. This property resembles that of collision-free TDMA protocols.

The virtual single-hop connectivity, however, strongly differentiates the LWB from existing TDMA approaches. In the LWB, the global communication schedule does not depend on the network topology. Given the total number of nodes in the network, slots are assigned based *only* on the (static or dynamically changing) application requirements, such as data rate or end-to-end delay. For each slot, the schedule specifies only the master node; all other nodes are implicitly slaves.

Inherent mobility support. When nodes gather information about the current network topology, they also need to adapt to changes in their environment. Mobility compounds the problem as topology changes frequently invalidate the information. Many protocols are thus designed under the assumption of static nodes, and would either perform rather poorly or not work at all when applied in a mobile setting [9].

The LWB supports mobility innately. In each flood, packet propagation does not depend on any knowledge of the network topology. As a result, *no changes are required to support mobile nodes*. Experiments in Sec. IV show that our LWB prototype achieves high data yield in static and mobile scenarios.

Inherent support for multiple sinks. Low-power wireless networks can effectively form the basis for closed-loop control systems [1]. In these systems, *multiple* sink nodes (actuators) must collect data generated by multiple source nodes. Tree-based routing designed for data collection at a *single* sink becomes insufficient, and several transport protocols are specifically targeted at these scenarios [20].

The LWB naturally supports multiple sinks. During each flood, all slaves receive with very high probability the packet transmitted by the master—all slaves are potential sinks. The application running at each node can decide whether to discard or use a packet, for example, to control an attached actuator.

III. PROTOTYPE

This section demonstrates the feasibility of the LWB by reporting on a prototype implementation and analyzing the QoS guarantees it provides. We evaluate the performance of this prototype in Sec. IV using testbed experiments. Our prototype is *one possible* incarnation of an LWB-based protocol, targeted at common application scenarios where nodes generate data with the same, fixed rate known prior to deployment [3]. We are currently working on more advanced incarnations of the LWB that do not require prior knowledge of the data rate, which can also vary among nodes and over time. Sec. V provides more details about our current work.

A. Implementation

We implemented a prototype of the LWB in Contiki for Tmote Sky nodes. As shown in Fig. 3 (A), communication over the LWB occurs within *communication rounds* repeated with period T . Nodes keep their radio off between two rounds to save energy. In periodic data collection, the round period T corresponds to the inter-packet interval at source nodes.

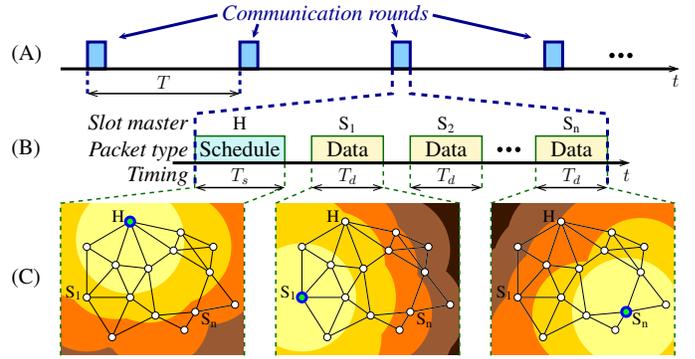


Figure 3. Communication over the LWB prototype occurs within periodic communication rounds (A). Each round consists of sequential communication slots (B), which correspond to distinct Glossy floods (C).

Fig. 3 (B) illustrates the sequence of actions during a single communication round. First, the host node H transmits the global communication schedule for the current round; this initial Glossy flood serves also to time-synchronize the nodes. The host transmits always the complete schedule, accounting already for future dynamic communication patterns with adaptive scheduling policies (see Sec. V). The time allocated for the host to transmit the schedule is $T_s = 20$ ms. The remaining part of the round is then divided into non-overlapping *communication slots* in which source nodes S_i send their application data. The master of a slot is determined by the schedule transmitted by the host. In each round, the host allocates for each source node one slot of constant length $T_d = 20$ ms.

Finally, as shown in Fig. 3 (C), each communication slot corresponds to a distinct Glossy flood, whose initiator (slot master) is determined by the schedule. Each Glossy flood naturally adapts to the current network topology, without requiring nodes to maintain any knowledge of it.

The LWB requires that at a certain time there is at most one initiator of a flood. Our implementation fulfills this requirement also in the presence of communication failures, which are unavoidable in wireless networks. Specifically, a node is allowed to initiate a flood *only* if it received the last message transmitted by the host, which provides time synchronization and the schedule for the current round; otherwise, it does not participate in any communication until it receives again a new schedule and thereby updates its synchronization status.

B. Provided QoS Guarantees

An important property of our LWB prototype is that every source node is assigned exactly one communication slot for each packet it needs to transmit every period T . In other words, there is a one-to-one correspondence between packets and slots: a packet is either received by the sink(s) within the respective slot or it is never received. As a result of its design, our LWB prototype provides the following QoS guarantees:

- *In-order packet delivery.* Packets arrive at the sink(s) in the same order in which they are transmitted by a source node.
- *Duplicate-free packet delivery.* If the reception succeeds, Glossy delivers a packet to the application running at the sink(s) only once at the end of a slot. Hence, the application receives packets transmitted by source nodes at most once.

- *Bounded end-to-end transmission delay.* The time between the first transmission of a packet and its successful reception at the sink(s) is bounded by the slot length T_d . A packet can only be received during its assigned slot, as the host never assigns additional slots for a lost packet.

As for the latter aspect, we acknowledge applications may also be interested in the latency from packet generation to successful reception. This is a specific facet of the scheduling algorithm, which we are considering in our current work along with other aspects, as we discuss in Section V.

IV. TESTBED EVALUATION

This section evaluates the performance of the LWB prototype using experiments on two wireless sensor testbeds.

A. Testbeds, Metrics, and Methodology

Testbeds. We use Twist [14] and our local testbed [4], [10]. The 85 Tmote Sky nodes available on Twist are densely spread among three floors in a university building. We set the transmit power to -7 dBm, resulting in a maximum distance of 3 hops among the nodes. Our local testbed consists of 43 Tmote Sky nodes distributed in several offices, passages, and storerooms on a single floor; two nodes are located outside on the rooftop. Using the highest transmit power of 0 dBm, nodes are at most 5 hops apart. On both testbeds, we use channel 26 to limit the interference with co-located WiFi. One node acts both as the host of the network and as the sink, collecting packets from the remaining nodes which periodically generate 36-byte packets.

Metrics. We evaluate the LWB prototype based on two standard performance metrics for data collection in low-power wireless networks [13]: (i) *data yield* is the fraction of packets successfully received by the sink; (ii) *radio duty cycle* is the fraction of time a node has its radio turned on.

Methodology. In scenarios with static nodes, we compare the performance of the LWB prototype with that of state-of-the-art data collection and link layer protocols. In particular, we use the most recent implementations of the Collection Tree Protocol (CTP) [13] and low-power listening (LPL) [21] available in TinyOS 2.1.1. For each scenario, we repeat the experiment three times with different LPL wake-up intervals: 50 ms, 200 ms, and 500 ms. All other parameters of CTP and LPL are kept at their default values. 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 to measure the energy efficiency of CTP and LPL.

B. Low data rate, static nodes

Scenario. We first consider a scenario typical of environmental monitoring [3]. These applications often use static nodes which generate periodic data samples at low rates. The main requirement is to achieve system lifetimes of several months or years, while providing a high data yield at the sink; long communication delays are typically tolerated [3]. To reproduce such scenarios, we let all 84 source nodes on Twist generate packets at a fixed inter-packet interval (IPI) of one minute. Thus, communication over the LWB repeats with round period

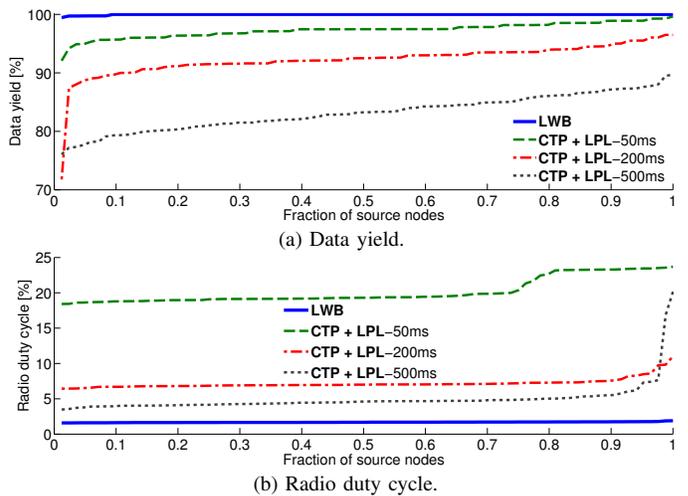


Figure 4. cdfs of the performance of the LWB and CTP + LPL on Twist. The LWB outperforms CTP + LPL for all settings, delivering 99.97 % of all packets to the sink at an average radio duty cycle of 1.69 % for IPI = 1 min.

$T = 60$ s. We perform several runs of 4 h each, with the LWB prototype and CTP on top of LPL (CTP + LPL).

Results. Fig. 4 plots cdfs of the performance of the LWB prototype and CTP + LPL. We see from Fig. 4(a) that the LWB achieves a data yield of 100 % for the majority of the nodes. The minimum data yield from a node is 99.45 %, the average data yield is 99.97 %. At the same time, as shown in Fig. 4(b), the LWB is highly energy-efficient. The energy consumption is evenly distributed among the nodes, with a maximum radio duty cycle of 1.90 % and an average of 1.69 %.

Moreover, the LWB outperforms CTP + LPL for all wake-up intervals we tested. The latter exhibit a clear trade-off between data yield and radio duty cycle. They achieve the highest average data yield of 97.31 % at a 50 ms wake-up interval, but this comes also at a very high average radio duty cycle of 20.18 %. Increasing the wake-up interval, however, results in a significantly lower data yield, averaging 83.19 % for a 500 ms wake-up interval. By contrast, the LWB is both reliable and energy-efficient, and requires only to set the round period T , which equals the IPI in periodic data collection.

C. Medium data rate, static nodes, wireless interference

Scenario. Next, we evaluate the robustness of the LWB and CTP + LPL against changes in the wireless channel conditions, which are common in low-power wireless networks [27]. We use the technique by Boano et al. [5] to generate controllable interference patterns, making the channel conditions vary in a repeatable manner. To this end, we add an interferer node to our local testbed in a position where it affects the communication of at least 10 nodes. When active, the interferer transmits a modulated carrier for 5 ms; then, it sets the radio to idle mode for 10 ms before transmitting the next carrier. The 42 source nodes transmit packets with IPI = 10 s. We use round period $T = 10$ s; all other settings of the LWB are identical to the previous scenario. The experiments last for 3 h, where the interferer is enabled in two periods of 30 min each.

Results. Fig. 5 shows the performance of the LWB prototype and CTP + LPL over time. We see that the LWB is very robust

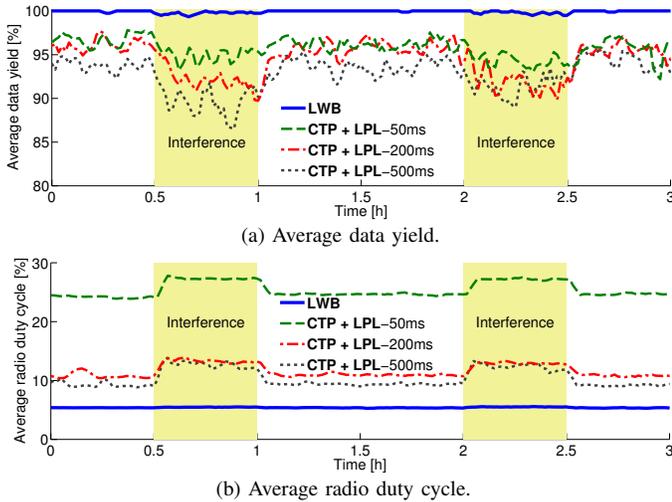


Figure 5. Performance of the LWB and CTP + LPL on our local testbed as the wireless channel conditions change. The LWB shows only marginal performance variations during periods of controlled wireless interference, whereas CTP + LPL decline considerably in both data yield and energy cost.

against changes in the channel conditions. The average data yield, shown in Fig. 5(a), experiences only marginal variations when the interferer is enabled, and the average radio duty cycle, shown in Fig. 5(b), remains almost constant. These are clear benefits of using Glossy as the only communication primitive for data collection. During each flood, Glossy adapts naturally to the current wireless channel conditions, without requiring any prior knowledge of them.

When the interferer is enabled, CTP tries to avoid low-quality links affected by interference and adapts the routing tree. This, however, leads to an increased contention over the remaining links, and thus to lower average data yield and higher average radio duty cycle (due to more retransmissions). This is especially evident with an LPL wake-up interval of 500 ms, given the very limited bandwidth this setting provides.

D. High data rate, mobile nodes

Scenario. We run an additional 3-hour experiment on our local testbed in which source nodes generate packets with $IPI = 1$ s. To account for the higher data rate, we adjust the round period to $T = 1$ s, without changing any of the remaining LWB settings. Moreover, we introduce mobility into the network by letting 6 of the 42 source nodes run on batteries and attaching them to people that wander around the testbed’s floor.

Results. Fig. 6(a) shows that a high data rate of one packet per second does not affect data yield in the LWB, which averages 99.74%. The same LWB prototype provides an equally high data yield at low data rates and high data rates. The goodput at the sink reaches 12 kbps.

The dark bars in Fig. 6(a) show that node mobility does not affect the LWB’s data yield either. This is even more evident in Fig. 6(b), which depicts the packet delivery of one mobile node during a time interval of 6 min. This node is carried back and forth by a person walking along the floor. The movement translates into a varying hop distance from the sink, measured based on the relay counter of the packet transmitted by the

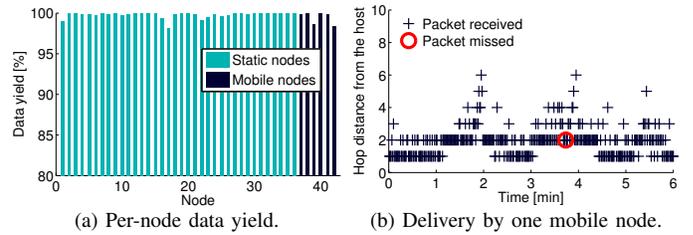


Figure 6. Packet delivery performance of the LWB in a connected network with a few mobile nodes and high data rate ($IPI = 1$ s). Because communication over the LWB is independent of the current network topology, it achieves an average data yield of 99.74% also when nodes are free to move.

sink [12]. During the short time interval, the sink misses only one of the 360 packets generated by the moving node.

These results clearly demonstrate the benefits of requiring no knowledge of the network topology for communication. In the LWB, the schedule depends only on the number of nodes and the application requirements (e.g., data rate); if these are fixed, so is the schedule. No changes in the implementation or the operational parameters are required to support mobility.

V. CURRENT WORK

Notwithstanding its limitations, the LWB prototype serves as a stepping stone for our future work, and helps us set a research agenda. This section describes our current work.

Scalability. Results from 85 nodes on Twist demonstrate that the LWB prototype is highly energy-efficient. Still, its energy consumption scales linearly with the total amount of traffic in the network. In order to apply the LWB to larger networks with hundreds or thousands of nodes, we are working on hierarchies of LWB’s. By assigning different wireless channels to disjoint sets of nodes, we group nodes into clusters (i.e., low-level buses), while several nodes from each cluster connect also to a shared high-level bus to exchange data among clusters.

Varying traffic demands. In many sensor network applications (e.g., event detection) the data rate depends on time-varying external stimuli. To support such applications, we are designing algorithms at the host that use traffic demands, embedded by nodes into data packets, to re-compute the schedule between rounds, and adapt the round period if necessary.

Dynamic set of nodes. Nodes may fail or be temporarily disconnected, for example, due to harsh environmental conditions [3]. As a result, the set of nodes in the network may change dynamically. We are applying policies to let the host detect when a node disappears, by keeping track of packets consecutively lost from a node. The host supports also new nodes joining the network, by scheduling at the end of a round additional contention slots without a predefined master.

QoS guarantees on reliability. The LWB prototype achieves data yield higher than 99.45% in all scenarios we tested. This high reliability is usually sufficient for applications that can tolerate (rare) packet loss. However, safety-critical applications may require QoS guarantees also on reliability. To this end, we are letting the host assign slots to the sink(s) at the end of a round for end-to-end acknowledgments; in case of missing acknowledgments, the host schedules additional slots for packet retransmissions during the next round.

Fault tolerance. To achieve a predictable behavior and provide QoS guarantees, the LWB employs a centralized approach where the host node orchestrates the entire communication. The major drawback of this solution is that a host failure is detrimental to the functioning of the system, similar to sink failures in tree-based routing protocols [7]. We are applying fault tolerance policies to replace a failed host with another (predefined) node. Compared to tree-based protocols, the LWB has the great advantage that all nodes receive all data with high probability: a new host node can immediately start distributing a schedule based on information it received before the failure.

VI. RELATED WORK

Providing QoS affects MAC and routing protocols as well as middleware and application [26]. The MiLAN middleware, for example, continuously adapts the network configuration based on the available network resources and the applications' QoS requirements and their relative importance [16]. As representatives of QoS-aware routing protocols, SAR maintains multiple paths to the sink and selects a path based on its energy resources and QoS, and the priority of a packet [23]; SPEED uses information about a node's one-hop neighborhood and geographic forwarding to find paths, while enforcing a uniform delivery velocity to bound the end-to-end packet delay [15]. Different from these solutions, the LWB takes a routing-free approach by mapping all communication onto Glossy network floods, and hence eliminates the need for duty-cycled link layers, routing protocols, neighbor tables, and link estimators.

The LWB is also related to work on TDMA-based communication protocols and scheduling algorithms. For example, WirelessHART, an open standard for industrial process monitoring and control, uses TDMA to approach deterministic communication [24]. Dozer represents a cross-layer solution comprising MAC, topology control, and routing, where nodes employ local time synchronization to schedule wake-ups among parents and children in the collection tree [6]. DRAND is a distributed, randomized TDMA slot assignment algorithm operating on a node's two-hop neighborhood [22]. The virtual single-hop connectivity provided by the LWB strongly differentiates it from existing TDMA approaches. While existing approaches allocate time slots to (possibly multiple non-interfering) sender-receiver pairs depending on topology and packet rates [11], the LWB requires no information about the network topology and computes a global schedule solely based on the application requirements, such as desired end-to-end communication delay and data rate.

VII. CONCLUSIONS

The LWB provides routing-free, QoS-aware data collection for low-power sensor networks. It achieves virtual single-hop connectivity among nodes in multi-hop networks by mapping all communication onto Glossy floods. Our LWB prototype provides bounded end-to-end delay and duplicate-free, in-order packet delivery, supports multiple sinks, and achieves energy-efficient and reliable operation under low and high traffic, static and mobile nodes, and wireless interference.

Acknowledgments. The authors thank David Hasenfratz and Olga Saukh for their feedback on early versions of this paper, and the anonymous reviewers for their helpful comments. This work was supported by Nano-Tera and NCCR-MICS, a center supported by the Swiss National Science Foundation under grant number 5005-67322.

REFERENCES

- [1] I. F. Akyildiz and I. H. Kasimoglu. Wireless sensor and actor networks: Research challenges. *Elsevier Ad Hoc Networks*, 2(4), 2004.
- [2] J. N. Al-Karaki and A. E. Kamal. Routing techniques in wireless sensor networks: A survey. *IEEE Wireless Commun. Mag.*, 2004.
- [3] J. Beutel et al. PermaDAQ: A scientific instrument for precision sensing and data recovery in environmental extremes. In *ACM/IEEE IPSN*, 2009.
- [4] J. Beutel et al. Poster abstract: The FlockLab testbed architecture. In *ACM SenSys*, 2009.
- [5] C. Boano, T. Voigt, N. Tsiftes, L. Mottola, K. Römer, and M. Zuniga. Making sensornet MAC protocols robust against interference. In *EWSN*, 2010.
- [6] N. Burri, P. von Rickenbach, and R. Wattenhofer. Dozer: Ultra-low power data gathering in sensor networks. In *ACM/IEEE IPSN*, 2007.
- [7] M. Ceriotti et al. Is there light at the ends of the tunnel? Wireless sensor networks for adaptive lighting in road tunnels. In *ACM/IEEE IPSN*, 2011.
- [8] J. I. Choi, M. Kazandjieva, M. Jain, and P. Levis. The case for a network protocol isolation layer. In *ACM SenSys*, 2009.
- [9] P. Dutta and D. Culler. Mobility changes everything in low-power wireless sensor networks. In *USENIX HotOS*, 2009.
- [10] M. Dyer et al. Deployment support network: A toolkit for the development of WSNs. In *EWSN*, 2007.
- [11] S. C. Ergen and P. Varaiya. TDMA scheduling algorithms for wireless sensor networks. *Springer Wireless Netw.*, 16, 2010.
- [12] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. Efficient network flooding and time synchronization with Glossy. In *ACM/IEEE IPSN*, 2011.
- [13] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. Collection tree protocol. In *ACM SenSys*, 2009.
- [14] 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.
- [15] T. He, J. Stankovic, C. Lu, and T. Abdelzaher. SPEED: A stateless protocol for real-time communication in sensor networks. In *IEEE ICDCS*, 2003.
- [16] W. Heinzelman, A. Murphy, H. Carvalho, and M. Perillo. Middleware to support sensor network applications. *IEEE Netw.*, 18(1), 2004.
- [17] A. Jhumka and L. Mottola. On consistent neighborhood views in wireless sensor networks. In *IEEE SRDS*, 2009.
- [18] M. Keller, L. Thiele, and J. Beutel. Reconstruction of the correct temporal order of sensor network data. In *ACM/IEEE IPSN*, 2011.
- [19] H. Kopetz. The time-triggered model of computation. In *IEEE RTSS*, 1998.
- [20] L. Mottola and G. P. Picco. MUSTER: Adaptive energy-aware multi-sink routing in wireless sensor networks. *IEEE Trans. Mobile Comput.*, 10(12), 2011.
- [21] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *ACM SenSys*, 2004.
- [22] I. Rhee, A. Warrier, J. Min, and L. Xu. DRAND: Distributed randomized TDMA scheduling for wireless ad-hoc networks. In *ACM MobiHoc*, 2006.
- [23] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie. Protocols for self-organization of a wireless sensor network. *IEEE Personal Commun. Mag.*, 7(5), 2000.
- [24] J. Song, S. Han, A. Mok, D. Chen, M. Lucas, M. Nixon, and W. Pratt. WirelessHART: Applying wireless technology in real-time industrial process control. In *IEEE RTAS*, 2008.
- [25] G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh. Fidelity and yield in a volcano monitoring sensor network. In *USENIX OSDI*, 2006.
- [26] Q. Zhang and Y.-Q. Zhang. Cross-layer design for QoS support in multihop wireless networks. *Proc. IEEE*, 96(1), 2008.
- [27] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *ACM SenSys*, 2003.