PaderMAC: A Low-Power, Low-Latency MAC Layer with Opportunistic Forwarding Support for Wireless Sensor Networks

Marcus Autenrieth and Hannes Frey

University of Paderborn
{marcus.autenrieth,hannes.frey}@upb.de

Abstract. Modern medium access control (MAC) protocols for wireless sensor networks (WSN) focus on energy-efficiency by switching a node's radio on only when necessary. This intoduced rendezvous problem is gracefully handled by modern asynchronous approaches to WSN MAC's, e.g. X-MAC, using strobed preambles. Nevertheless, most MAC layer ignore the possible benefits in energy consumption and end-to-end latency, supporting opportunistic routing can provide. In this paper we present PaderMAC, a strobed preamble MAC layer which supports cross-layer integration with an arbitrary opportunistic routing layer. This work specifies the PaderMAC protocol, explains its implementation using TinyOS and the MAC layer architecture (MLA), and presents the results of a testbed performance study. The study compares PaderMAC in conjunction with opportunistic routing to X-MAC in conjunction with path-based routing and shows how PaderMAC reduces the preamble length, better balances the load and further improves the end-to-end latency within the network.

1 Introduction

Replacing batteries is impossible or infeasible in many deployments of wireless sensor networks (WSNs). For this reason, reducing the power consumption of WSN nodes is a central research topic for prolonging the lifetime of such a network. The most common technique to reduce the power consumption on nodes is to let them *sleep*, i.e. power down their radio, most of the time and let them *wake up*, i.e. switch their radio on, only when needed to send or receive packets. This approach, called duty-cycling, introduces a problem, called the *rendezvous problem*, which the medium access control (MAC) layer must solve: transmitting and receiving node must both be awake in order to have a successful transmission. Solving this rendezvous problem in an energy-efficient manner is one of the key focuses of WSN MAC design.

So far, most MAC and routing layers for wireless sensor networks have been developed without considering the possible performance gains and implementational issues of a practical cross-layer integration on real hardware.

In this paper we introduce PaderMAC, a sender-initiated, asynchronous MAC layer which extends X-MAC [4] with opportunistic forwarding support. Section 2

H. Frey, X. Li, and S. Ruehrup (Eds.): ADHOC-NOW 2011, LNCS 6811, pp. 117–130, 2011.

will present the related work, giving a short summary on opportunistic routing mechanisms and sender-initiated asynchronous MAC layer approaches. Section 3 will first motivate with a sink-based routing scenario why opportunistic routing support is a valuable extension for X-MAC. It will then explain how PaderMAC integrates with an opportunistic routing layer to address a set of eligible recipients during a transmission and how collisions between contending recipients are handled. It will also explain how PaderMAC retains backwards-compatibility with X-MAC after motivating why this is desireable.

Section 4 will then cover the reference implementation of PaderMAC in TinyOS [13], using the MAC layer architecture (MLA) [14]. Its focus lies on the cross-layer integration with an opportunistic routing layer, but it also includes the contribution of a feedback-enabled preamble-sender component to the MLA. Section 5 will then show how opportunistic routing using PaderMAC can reduce energy consumption and end-to-end latency in comparison to routing using X-MAC. Section 6 then summarizes these findings and points to future research directions.

2 Related Work

The MAC layer and the routing layer integration described in this work is related to two research areas, routing mechanisms which support opportunistic packet forwarding and preamble sampling based duty cycling MACs. Both areas are summarized in the following two sections.

2.1 Routing Mechanisms Supporting Opportunistic Packet Forwarding

Unicast Communication. In opportunistic packet forwarding the next hop node is not decided at the time of message transmission but depends on which of the nodes in vicinity could successfully receive that transmission. This technique was exploited in the seminal opportunistic forwarding schemes ExOR [2] to increase network throughput, and GeRaF [30] to reduce network latency and energy consumption. Both schemes employ a greedy strategy to decide which node is eligible to forward a received message.

Greedy forwarding schemes, which were not considered as an opportunistic forwarding strategy when introduced, have already been described in earlier work [7,26,18,24]. In those schemes the next hop node is selected depending on the position of that node and the position of the destination node. In principle, however, any node geographically closer to the destination than the current node is a potential forwarder. This allows an opportunistic realization of these greedy routing schemes which has been considered by the so called beaconless greedy routing variants BLR [12], IGF [3], and CBF [11].

A further geographic forwarding mechanism which supports opportunistic packet forwarding is given by the geographical clustering idea [10,9,27,20]. Using a regular partitioning of the space, each node is assigned to the cluster it is located in. Packet forwarding is done on geographic cluster level, i.e., a forwarder's task is to reach any node in a neighboring cluster but not necessarily a specific one. In the same way as done with the beaconless greedy routing variants this allows an opportunistic forwarding realization.

In the listed forwarding schemes the receiving nodes utilize information about geographic node locations to decide if they are eligible for forwarding. However, even without geographic location information, opportunistic forwarding variants are possible. Examples for unicast communication are link-reversal based routing schemes like TORA [19].

Sink-oriented Communication. Opportunistic forwarding is of course not limited to unicast communication. In sink-oriented communication, sensor nodes use intermediate nodes to transmit their measurement data towards one or a set of dedicated sink nodes.

The CTP protocol [8] is an example of such a sink-oriented communication protocol, which could be modified to allow opportunistic packet forwarding. With initial flooding and a sophisticated repair mechanism, distance values from the nodes towards sink nodes are maintained. Using these values, each node can compare its own distance with the distance of its neighbor and thus opportunistically decide if it is eligible to forward the message.

2.2 MAC Layers for Wireless Sensor Networks

MAC layers for wireless sensor networks can be classified into three major directions of approach: synchronous, asynchronous, and hybrid approaches.

In the synchronous approaches [29,28,17,16], nodes form groups whose members all sleep and wake up simultaneously. This incurs some overhead, as a master node has to be elected, which has to distribute a sleep-wakeup-schedule for its group and which has to ensure that all members keep synchronized.

In the asynchronous approaches nodes follow their sleep cycles notwithstanding the other nodes' ones. When a message has to be transmitted, sender and receiver have to solve the rendezvous problem on demand. Two ways can be followed here: a *sender-initiated* and a *receiver initiated* approach. In the sender-initiated approach [21,4], the sender announces a pending transmission by first sending a preamble. Once the sender has ensured that the receiver is awake, it transmits the data frame. In contrast the receiver initiated approach is to let the receiver announce a short beacon when it wakes up. The sender has to monitor the channel and can start its transmission once it receives a beacon from the intended receiver node [25].

The hybrid approaches [1,6,22] combine synchronous and asynchronous techniques to mitigate weaknesses or tune a MAC protocol to a specific scenario.

In the domain of sensor networks an asynchronous sender initiated MAC was initially introduced with the B-MAC protocol [21]. In B-MAC, a sender starts its transmission by sending a preamble after performing a clear channel assessment (CCA). A node which overhears the preamble stays awake and waits for the sender to transmit the data frame. Since the preamble must ensure that the designated recipient is awake, is has to be transmitted for a complete sleep

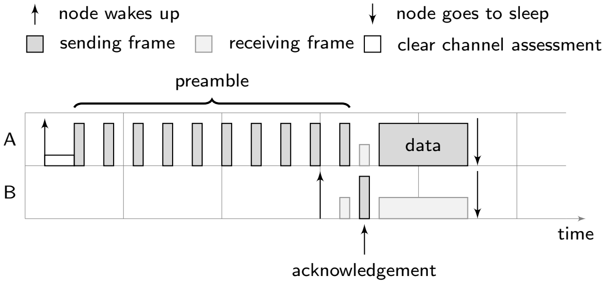

Fig. 1. An example for an X-MAC transmission

period, before the data-frame is transmitted. Clearly, this wastes energy on both, sender and receiver.

like illustrated in Figure 1, X-MAC [4] addressed this issue of B-MAC, by replacing the bitstream preamble of B-MAC by request-to-send (RTS) frames, which are sent repeatedly up to a duration of a sleep period. In between the RTS frames, the receiver can now reply with a clear-to-send (CTS) frame, also called an "early acknowledgement" (early ACK), to cut the preamble short. Upon receiving a CTS, the sender immediately sends the data frame and both nodes can go back to sleep. Since the RTS frames contain the address of the intended receiver, every other node which overhears an RTS can go directly back to sleep. This reduces the duration the sender has to transmit its preamble, shorten By shortening the duration a sender has to transmit its preamble, as well as the overall transmission, energy consumption is reduced.

3 PaderMAC: A WSN MAC Protocol with Opportunistic Forwarding Support

In contrast to previous simulative and analytical approaches [15,23], PaderMAC is a practical MAC protocol with opportunistic forwarding support, compatible with modern sensor node hardware, such as the Tmote-Sky platform. By design, it is not tailored to a specific routing mechanism, but tries to be usable with several opportunistic and non-opportunistic routing protocols.

The central idea behind PaderMAC is to improve a network's lifetime by further shortening energy-consuming preambles. This is achieved by exploiting the fact that, in a sufficiently dense multi-hop network, multiple relays for a given destination can be found in each hop.

As an example, consider a sink-based routing scenario, commonly used in data-centric WSN deployments. The traditional approach is to create and maintain a data-gathering tree, using a distributed algorithm and an appropriate metric, e.g.: hop-count or ETX [5], which assigns to each node a relay-node, responsible for forwarding the data.

Figure 2 shows an example with node B acting as the sink. Packets from node A to node B are routed along the predefined path [A → 5 → 2 → sink]. In

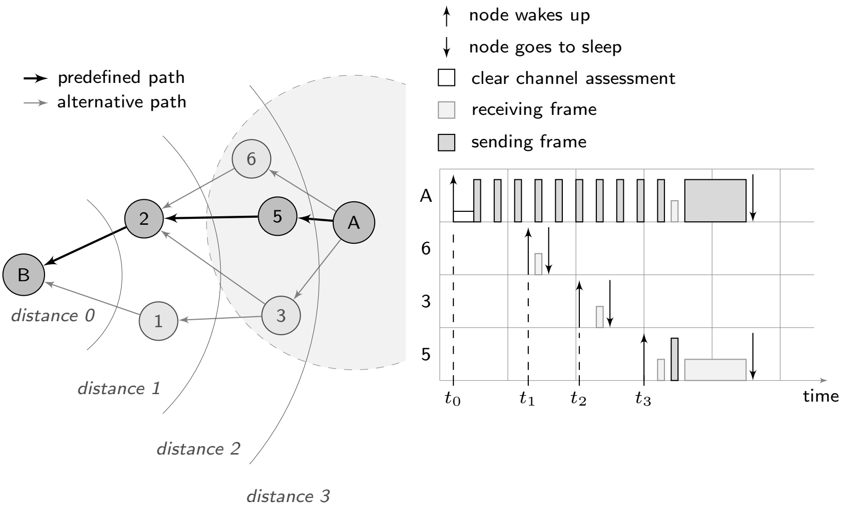

Fig. 2. An example for energy saving opportunities, by using alternative paths in multihop end-to-end transmissions

addition three alternative paths exist: [A → 6 → 2 → sink], [A → 3 → 2 → sink] and [A → 3 → 1 → sink].

At time t_0, Node A wakes up and wants to transmit data towards node B. In this example node 5 is used as a fixed relay, though nodes 6 and 3 could also act as a relay. After performing a CCA, node A starts its preamble, waiting for the rendezvous with node 5. At time t_1 node 6 and at time t_2 node 3 wake up, but each goes back to sleep immediately since they both are not addressed in the preamble of node A. Finally node 5 wakes up at time t_3 and receives the packet. By handing over the packet to node 6, instead of node 5, the preamble could be shortened by a duration of $t_3 - t_1$, thus saving additional energy on the sender.

3.1 Opportunistic Routing Integration

In principle, the opportunistic routing support of PaderMAC works as follows: A sender embeds the routing information into its preamble, and based on that information, a receiver decides whether it is a suitable recipient for the transmission, i.e. whether it can successfully forward the data to its final destination. If more than one recipient contends for the packet, a receiver contention mechanism (described in section 3.2) is used to resolve the conflict.

To keep PaderMAC as routing-agnostic as possible, the decision whether to contend for a transmission or not, has to be encapsulated by the routing-layer. When a preamble frame announcing an opportunistic transmission is received, PaderMAC hands that frame for inspection to the routing layer, which responds with its decision. The inter frame spacing of the opportunistic routing preambles have to be adapted to the complexity of the decision, made by the routing layer.

Not every packet transmission needs opportunistic forwarding. For example, it might be necessary for the routing layer to send additional management packets like beacons, or send packets over a predefined path in case the opportunistic forwarding fails. To distinguish between opportunistic and normal transmission, PaderMAC employs two differenc types of preamble-frames.

3.2 The Receiver Contention Mechanism

During an opportunistic forwarding transmission, it can happen that two or more suitable forwarders wake up and contend for the transmission. Several circumstances prevent the receiver contention from being handled with a simple carrier sense multiple access (CSMA) scheme. The first one is the fact that PaderMAC sends its preamble frames in a tight loop with very small gaps in between, which are used for the early ACK. Increasing these gaps is infeasible, since it would result in increased listen-periods for nodes which wake up, thus increasing their energy consumption. Another reason against a CSMA scheme is the possibility that two contending receivers could be subject to a hidden terminal problem, a situation where a carrier sense is ineffective for avoiding collision since the contending nodes are unable to detect each other's transmission.

PaderMAC does not try to prevent the inevitable, and instead employs a slotted backoff mechanism for collided acknowledgements. Its backoff mechanism exploits the fact, that the sender does not recognize collided early ACKs, since both ACKs destroy each other at the sender. Because it cannot detect the collision the sender then proceeds with sending its preamble. A receiver which overhears a preamble frame *after* sending an early ACK assumes that a collision took place and switches into backoff-mode. It randomly picks a number of additional preamble frames it has to receive from the sender before it resends its early ACK. As soon as the sender receives the first early ACK from a contending receiver, it transmits the data frame addressed to that receiver. To identify the receiver which won the contention, receivers need to identify themselves to the sender. This is done by each receiver embedding its MAC address in its early ACK. The sender then uses this information to address the data frame to the receiver which won the contention.

Figure 3 shows an example situation for the receiver contention mechanism of PaderMAC. In the example the two receivers, node A and B wake up almost simultaneously and since they are both hidden from each other they cannot sense each others start of transmission and their acknowledgements collide. Both acknowledgements get destroyed at the sender node S and it continues to send its preamble. Upon overhearing a preamble frame after sending an ACK, both nodes switch into backoff mode. In the example A choses a backoff-window of two preamble frames and B a backoff-window of 3. Under the assumption that both nodes can receive each following preamble, node A will win the contention. But now let's assume in this exmample that node A only receives one of the subsequent preambles. Since a contending node can only count preamble frames it successfully receives and decodes, node A has to wait until the channel quality improves to a point where it can successfully receive the remaining

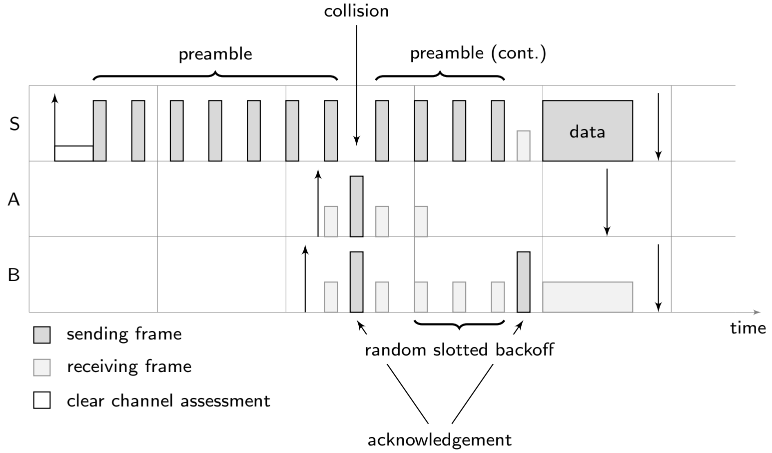

Fig. 3. An example for collision handling in PaderMAC

preamble frames or another node wins the contention or the transmission times out. Because of this effect, the contention mechanism is slightly biased towards contenders with low packet error rates on the downlink.

4 Implementation

We implemented PaderMAC on TmoteSky hardware using TinyOS [13] and the MAC Layer Architecture (MLA) [14]. Since it uses the MLA's implementation of X-MAC as a starting point, PaderMAC superficially resembles X-MAC in its architecture, though important parts work much differently. In contrast to the MLA implementation of X-MAC, PaderMAC uses a feedback-capable component for sending a preamble and needs a sophisticated component for handling the receiver contention mechanism for opportunistic forwarding transmissions.

Like illustrated in Section 3, PaderMAC needs the early ACKs, sent by the contending receivers, to contain the MAC address of the contenders. The preamble-sender component as provided by the MLA, could not be used, since it's implementation relies on the low-level acknowledgements of TinyOS. We therefore designed a new component for sending preambles, called SoftAckPreambleSenderC. The SoftAckPreambleSenderC component itself is designed for the development of preamble-based MAC layers which need to stop their preamble based on some arbitrary event. This is done by decoupling the standard PreambleSenderC component from the low-level radio interface via the SoftAckHandler interface.

In contrast to the straightforward receiver mechanism of X-MAC, the receiver contention mechanism, encapsulated in the component PaderMACReceiverEngineP, plays a central role in PaderMAC during opportunistic forwarding transmissions. It has to communicate the reception of an ACK to the SoftAckPreambleSenderC component, it has to receive and process the two types of preamble frames and it has to communicate with the routing layer in order to decide whether or not to contend for an opportunistic forwarding transmission.

To integrate PaderMAC with an opportunistic routing layer, we defined the RoutingDecider interface. The interface specifies the `do_forward` function, which accepts a preamble frame as its argument and returns a boolean value, indicating the routing layers decision. An opportunistic routing layer must now implement this interface, to encapsulate the opportunistic routing decision, and wire it to PaderMAC's MacControlC component. That way, PaderMAC can hand over the opportunistic forwarding decision to the routing layer. This handover, makes PaderMAC very flexible, but comes at the cost of having to manually tune the ACK-timeouts of the SoftAckPreambleSenderC component to the computation time of the component providing the RoutingDecider interface.

5 Performance Study

This section describes the conducted measurements to evaluate the performance of PaderMAC versus X-MAC in terms of ernergy consumption and end-to-end-latency. Our measurements have a clear focus on the MAC-layer performance, sacrificing some realism on the routing layer for a more transparent understanding of the MAC behaviour.

The goal of our performance study is to get an estimate of the performance of PaderMAC vs. X-MAC in a low-traffic, multihop scenario, without cross-traffic. For the spatial setup, we created an equidistand grid of TmoteSky sensor nodes under our laboratory's ceiling, with 6 rows and 5 columms.

5.1 Routing Scheme

Our measurements aim to study the performance of PaderMAC vs. X-MAC under almost routing-agnostic conditions.

Our measurements use geo-routing on the grid coordinates to send a packet on a round-trip from position (1,1) to (5,6) and back. This enabled us to measure the end-to-end latency without having to perform tedious clock-synchronization for being resilient to clock-drift. The performance of PaderMAC is measured using a greedy packet forwarding scheme. X-MAC (emulated by the legacy support of PaderMAC) is measured by routing the packets along the shortest path through the grid. Both routing methods were integrated into one routing layer, to speed up the measurement.

5.2 Scenario Description

To model an asynchronouse wakeup-pattern during our measuremets, the duty-cycles of the motes were jittered randomly using an uniform distribution over the sleep period. This method was also used to model the random generation of packets on the source mote at position (1,1).

The only varied parameter in the measurement was the sleep period, which ranged from 1000 ms to 4500 ms, increasing in steps of 500 ms. For each sleep period 30 samples for each MAC protocol were measured, resulting in overall 240 samples. The order in which those samples were measured was scrambled in order to decorrelate the results in time. In each sample, for each MAC layer, we sent 5 consecutive packets across the network.

5.3 Results

In our multihop measurement, we were interested in three metrics: power-consumption, latency and fairness. All confidence intervals have a confidence-level of 95%.

Figure 4, presents the latency comparison, where PaderMAC clearly dominates in the domain of larger sleep periods.

Interestingly X-MAC compares quite well for smaller sleep periods, because it always uses the shortest path (in hops) across the network, while PaderMAC can opportunistically choose shorter hops, which lead to longer pathes, impacting the end-to-end latency.

Nevertheless, with increasing sleep periods, when X-MAC has to wait longer for the forwarders to wake up, PaderMACs opportunistic forwarder selection pays off.

Since sending the preamble consumes most energy, we measured the preamble durations for each transmission, but also had to account for the different path length, i.e. different number of single transmissions. We therefore took the sum of all preamble durations for each sample to have a fair comparison in energy consumption. Using the CC2420 datasheet we computed the energy consumption in Joule, used for all preambles in and end-to-end transmission.

As shown in Figure 5, PaderMAC can save up to a factor of two in energy consumption, compared to X-MAC.

In addition to the above mentioned metrics, we measured how fair the task of relaying packets towards the destination and hence the energy-consumption was distributed over the available relays. The fairness shown in Figure 6 results from computing the Herfindahl index over the number of packets each relay-node received. With $N = 28$ being the number of relay nodes and r_i being the number of packets relay i received, the Herfindahl index H is computed as

$$H := \frac{\sum_{i=1}^{N} r_i^2}{(\sum_{i=1}^{N} r_i)^2} \ . \tag{1}$$

Since all nodes, except those at position (1,1) and (5,6) act as a relay, the ideal fairness for our multihop scenario is 1/28. Figure 7 illustrates how the load of forwarding packets was distributed over the network during our measurement, with the hotspots illustrating the shortest path, taken by X-MAC and showing the smooth load distribution of PaderMAC, albeit with the possibility of slightly longer pathes.

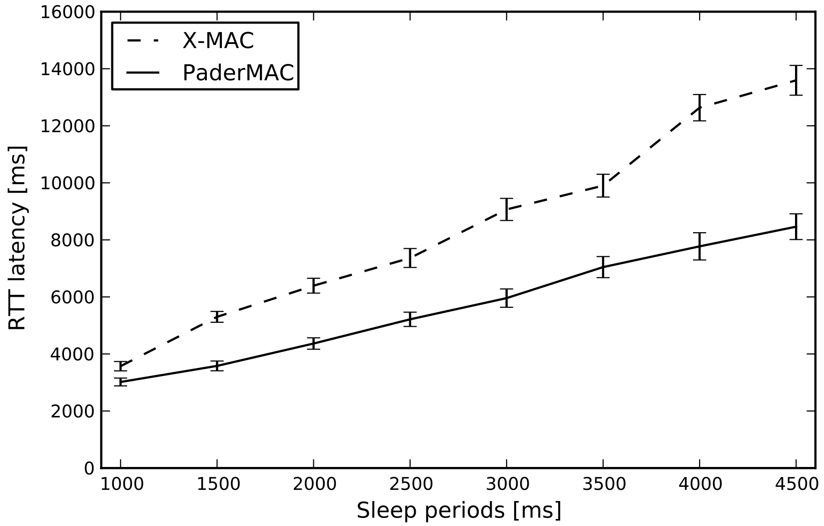

Fig. 4. The end-to-end latency for a roundtrip from position (1,1) to (5,6) and back

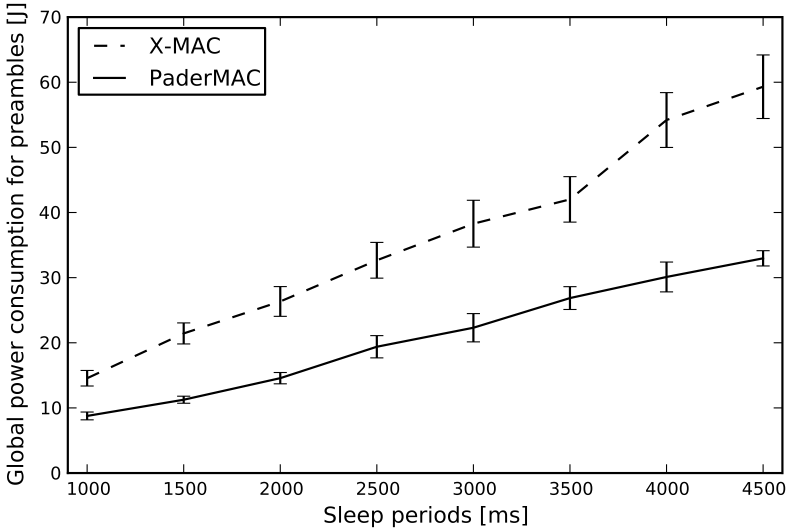

Fig. 5. The power consumption, measured by adding up the preamble-duration along the path

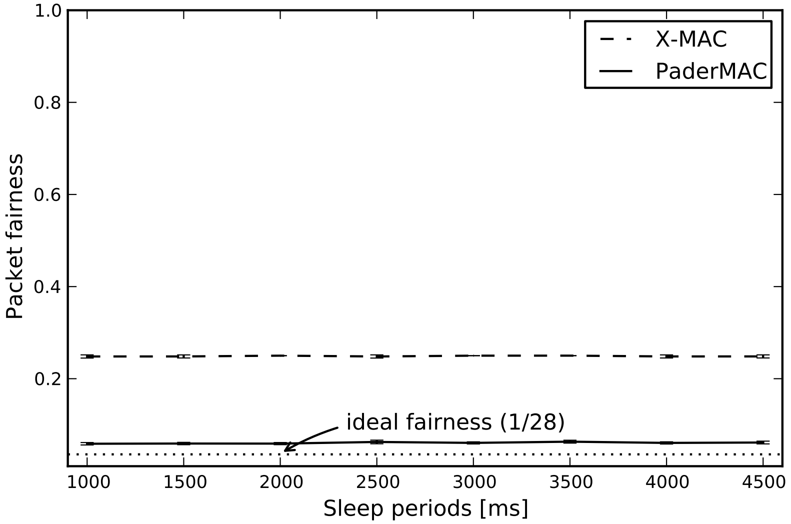

Fig. 6. Load-balancing in terms of packet-fairness

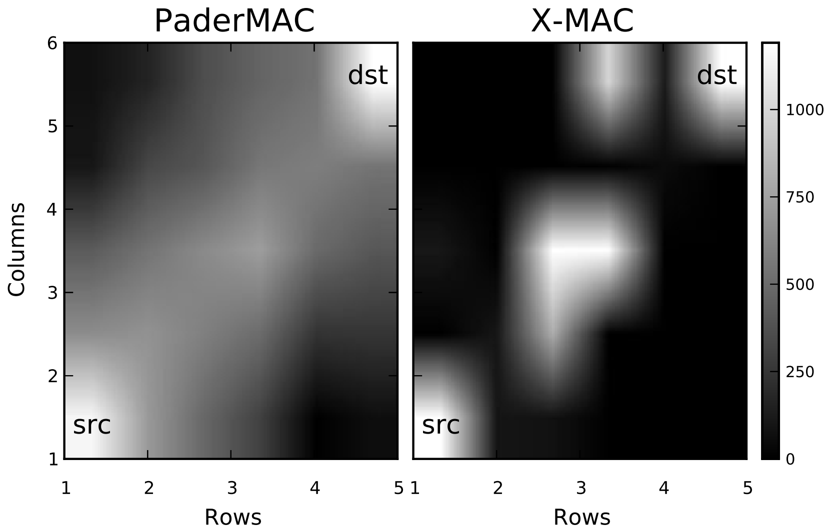

Fig. 7. Heat map of forwarded packets throughout the node grid

The near ideal fairness of PaderMAC, as shown in Figure 6, in conjunction with the reduced power consumption, shown in Figure 5 can drastically reduce the mean time to node failure in sensor networks.

6 Conclusions

Integrating a strobe preamble MAC layer with an opportunistic routing layer is an appealing cross layer optimization approach. This work describes PaderMAC, a practical implementation of such a MAC layer integration using the runtime environment TinyOS and the MAC layer architecture MLA. Our implementation contributes an extension for the MLA, a feedback-capable component for sending preamble strobes. The MAC protocol design of PaderMAC is generic in that respect that it can also be combined with other opportunistic routing layer implementations. And of course by combining it with a traditional non-opportunistic routing layer implementation the functionality of X-MAC is contained in the PaderMAC implementation as well.

Our real hardware testbed evaluation of X-MAC and PaderMAC in conjunction with a simple geographic greedy routing mechanism complements already known theoretical and simulation studies on integrating a strobe preamble MAC layer with opportunistic routing. It illustrates in practice how opportunistic routing in conjunction with such MAC layer can improve the performance of WSNs in terms of reduced end-to-end latency, reduced power consumption and increased fairness.

Two future research tracks can be followed with respect to PaderMAC. On one side PaderMAC can be combined and studied with routing schemes which are more sophisticated than plain opportunistic greedy packet forwarding. In particular, an integration of PaderMAC with CTP, allowing nodes improving progress toward the sink or a set of sink nodes to contend for packet forwarding is an interesting WSN relevant research direction.

A further research track is a PaderMAC extension improving how the next relay node is selected. So far the relay for each transmission is implicitly selected via the receiver contention mechanism: the earliest relay or the relay with the smallest backoff window wins. A future PaderMAC implementation could however let the preamble sender first collect several potential receiver replies and then decide the best one out of these (for instance the receiver minimizing the hop distance or the ETX towards the end receiver). How long the sender should wait for possibly better next hop node replies then becomes a classical optimal stopping problem.

References

1. Ahn, G., Hong, S., Miluzzo, E., Campbell, A., Cuomo, F.: Funneling-mac: a localized, sink-oriented mac for boosting fidelity in sensor networks. In: Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, p. 306. ACM, New York (2006)
2. Biswas, S., Morris, R.: Exor: Opportunistic multi-hop routing for wireless networks. In: Proceedings of the Annual Conference of the Special Interest Group on Data Communication (SIGCOMM), pp. 133–144 (2005)

3. Blum, B.M., He, T., Son, S., Stankovic, J.A.: IGF: A state-free robust communication protocol for wireless sensor networks. Tech. Rep. CS-2003-11, Department of Computer Science, University of Virginia (April 21, 2003)
4. Buettner, M., Yee, G., Anderson, E., Han, R.: 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, p. 320. ACM, New York (2006)
5. De Couto, D.S.J., Aguayo, D., Bicket, J., Morris, R.: A high-throughput path metric for multi-hop wireless routing. In: Proceedings of the 9th Annual International Conference on Mobile Computing and Networking, p. 134. ACM Press, New York (2003)
6. El-Hoiydi, a., Decotignie, J.D.: WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks. In: Proceedings of Ninth International Symposium on Computers and Communications, ISCC 2004 (IEEE Cat. No.04TH8769), vol. 1, pp. 244–251 (2007)
7. Finn, G.G.: Routing and addressing problems in large metropolitan-scale internetworks. Tech. Rep. ISI/RR-87-180, Information Sciences Institute (ISI) (March 1987)
8. Fonseca, R., Gnawali, O., Jamieson, K., Levis, P.: Collection tree protocol. In: SenSys 2009: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems. ACM, New York (2009)
9. Frey, H.: Geographical cluster based routing with guaranteed delivery. In: 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2005), Washington, DC, USA (November 7-10, 2005)
10. Frey, H., Görgen, D.: Geographical cluster based routing in sensing-covered networks. IEEE Transactions on Parallel and Distributed Systems: Special issue on Localized Communication and Topology Protocols for ad hoc Networks 17(4), 885–891 (2006)
11. Füßler, H., Widmer, J., Käsemann, M., Mauve, M., Hartenstein, H.: Contention-based forwarding for mobile ad-hoc networks. Ad Hoc Networks 1(4), 351–369 (2003)
12. Heissenbüttel, M., Braun, T.: BLR: Beacon-less routing algorithm for mobile ad-hoc networks. Elsevier's Computer Communications Journal 27, 1076–1086 (2003)
13. Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D., Pister, K.: System architecture directions for networked sensors. ACM Sigplan Notices 35(11), 93–104 (2000)
14. Klues, K., Hackmann, G., Chipara, O., Lu, C.: A component-based architecture for power-efficient media access control in wireless sensor networks. In: Proceedings of the 5th International Conference on Embedded Networked Sensor Systems - SenSys 2007, vol. 1, p. 59 (2007)
15. Lin, E.Y., Rabaey, J., Wolisz, A.: Power-efficient rendez-vous schemes for dense wireless sensor networks. In: Proceedings of the IEEE International Conference on Communications, vol. 7, pp. 3769–3776 (June 2004)
16. Lin, P., Qiao, C., Wang, X.: Medium access control with a dynamic duty cycle for sensor networks. In: IEEE Wireless Communications and Networking Conference, WCNC 2004, vol. 3, pp. 1534–3159 (2004)
17. Lu, G., Krishnamachari, B., Raghavendra, C.S.: An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks. In: Proceeings of the 18th International Parallel and Distributed Processing Symposium (2004)
18. Nelson, R., Kleinrock, L.: The spatial capacity of a slotted aloha multihop packet radio network with capture. IEEE Transactions on Communications 32(6), 684–694 (1984)

19. Park, V.D., Corson, M.S.: A highly adaptive distributed routing algorithm for mobile wireless networks. In: Proceedings of the 16th IEEE Conference on Computer Communications (INFOCOM 1997) (1997)
20. Philip, S.J., Ghosh, J., Ngo, H.Q., Qiao, C.: Routing on overlay graphs in mobile ad hoc networks. In: Proceedings of the IEEE Global Communications Conference, Exhibition & Industry Forum (GLOBECOM 2006) (2006)
21. Polastre, J., Hill, J., Culler, D.: Versatile low power media access for wireless sensor networks. In: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, pp. 95–107. ACM, New York (2004)
22. Rhee, I., Warrier, A., Aia, M., Min, J., Sichitiu, M.L.: Z-MAC: A Hybrid MAC for Wireless Sensor Networks. IEEE/ACM Transactions on Networking 16(3), 511–524 (2008)
23. Shah, R., Wietholter, S., Wolisz, a., Rabaey, J.: When Does Opportunistic Routing Make Sense? In: Third IEEE International Conference on Pervasive Computing and Communications Workshops, vol. 1, pp. 350–356 (March 2005)
24. Stojmenovic, I., Lin, X.: Power-aware localized routing in wireless networks. IEEE Transactions on Parallel and Distributed Systems 12(11), 1122–1133 (2001)
25. Sun, Y., Gurewitz, O., Johnson, D.B.: RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks. In: SenSys 2008: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, pp. 1–14. ACM, New York (2008)
26. Takagi, H., Kleinrock, L.: Optimal transmission ranges for randomly distributed packet radio terminals. IEEE Transactions on Communications 32(3), 246–257 (1984)
27. Tejeda, H., Chávez, E., Sanchez, J.A., Ruiz, P.M.: A virtual spanner for efficient face routing in multihop wireless networks. In: Cuenca, P., Orozco-Barbosa, L. (eds.) PWC 2006. LNCS, vol. 4217, pp. 459–470. Springer, Heidelberg (2006)
28. Van Dam, T., Langendoen, K.: An adaptive energy-efficient MAC protocol for wireless sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, p. 180. ACM, New York (2003)
29. Ye, W., Heidemann, J., Estrin, D.: An energy-efficient MAC protocol for wireless sensor networks. Tech. rep., USC/ISI (2001)
30. Zorzi, M., Rao, R.R.: Geographic random forwarding (geraf) for ad hoc and sensor networks: Energy and latency performance. IEEE Transactions on Mobile Computing 2(4), 349–365 (2003)