Medium Access Control
Facing the Reality of WSN Deployments

Romain Kuntz, Antoine Gallais and Thomas Noel
Image Sciences, Computer Sciences and Remote Sensing Laboratory (LSIIT UMR CNRS 7005)
University of Strasbourg, France
{kuntz,gallais,noel}@unistra.fr

This article is an editorial note submitted to CCR. It has NOT been peer reviewed. The author takes full responsibility for this article's technical content. Comments can be posted through CCR Online.

ABSTRACT

Although research on algorithms and communication protocols in Wireless Sensor Networks (WSN) has yielded a tremendous effort so far, most of these protocols are hardly used in real deployments nowadays. Several reasons have been put forward in recent publications. In this paper, we further investigate this trend from a Medium Access Control (MAC) perspective by analyzing both the reasons behind successful deployments and the characteristics of the MAC layers proposed in the literature. The effort allocated to develop suitable protocols from scratch every new deployment could however be minimized by using already existing contributions which provide code reuse and adaptive protocols. Though we advocate their use for nowadays deployments, we have identified several shortcomings in foreseen scenarios for which we provide guidelines for future researches.

Categories and Subject Descriptors

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

General Terms

Design

Keywords

Wireless Sensor Networks, Survey, Deployments, MAC

1. INTRODUCTION

Natural disaster prevention [1], wildlife tracking [2] or structural health monitoring [3] are some of the appealing promises made by Wireless Sensor Networks (WSN). This vast range of applications must operate in a very restricted environment in terms of energy, computation and memory. Even though this makes the medium access complex, a number of effective deployments have been reported in the last decade. We will detail in Section 2 their characteristics and the main reasons of their successful achievements.

In parallel, the new challenges raised by WSN have naturally attracted the interest of the research community, which has proposed significant improvements within the communication stack of the sensor nodes [4]. However, this wide variety of protocols is barely used in real deployments of WSN. An analysis of various deployments and protocols [5] has raised three major reasons for this issue. First, protocols from the literature do not describe with enough precision the applications for which they are designed. Then, they usually neglect very simple design choices and prefer complexity. Finally, their evaluations fail to match the hypothesis of real deployments. In this paper, we will further study these observations from the Medium Access Control (MAC) perspective. Especially, the analysis presented in Section 3 reveals the difficulty to exploit roughly specified MAC protocols in real deployments.

This situation often instigates engineers to define their own protocol stack from scratch [6], which prevents non-expert users from having access to optimized designs. Flexible protocols and code reuse would however considerably alleviate the development phase prior to a deployment. Among the already existing contributions, we will summarize in Section 4 the ones that could be used to ease the design of a MAC layer if a WSN were to be deployed nowadays.

With the foreseen adoption of dynamic scenarios which involve mobility or multiple data collection schemes, the design of MAC protocols will become increasingly complex. We will try to identify in Section 5 the missing pieces as well as provide guidelines for future researches in this area. Concluding remarks will follow in Section 6.

2. OVERVIEW OF WSN DEPLOYMENTS

2.1 Main Characteristics

Among the various deployments outlined in the literature, we have tried to characterize their applications from the MAC layer perspective. Especially, we have classified design choices and constraints that could have a direct impact on the MAC protocol. We could corroborate the statement that is briefly raised in [3]: the nature of the data traffic reveals two major and distinct categories of applications in WSN.

The first one gathers deployments that aim at continuously monitoring a phenomenon: habitat [2] and glacier [6] monitoring belong to this category. The sensor nodes report their measured values in a time-driven fashion. This data collection scheme usually mandates the nodes to operate with low duty-cycle and low power consumption, a small sampling rate and low data rates.

The second category covers deployments intended for measurement of a response to stimuli. This includes for example structural health monitoring [3]. A query-driven or event-driven data collection scheme is more suitable for this category of applications than a time-driven one. As a mat-

Characteristics	*Today's deployments*	*Impact on the MAC layer*
Tolerance to delay	High tolerance	Best effort schemes are adequate
Size of the network	Relatively small	Small collision domain
Topology	Single to few hops	Simplified link management
Sensor placement	Carefully studied	Simplified neighborhood management
Length of deployment	Less than a year	Energy management is a secondary concern
Mobility	Limited to a few number of nodes	Simplified scheduling scheme

Table 1: Main characteristics of WSN deployments nowadays

ter of fact, these applications require high data rates, high fidelity sampling (through a reliable end-to-end protocol), precise time-stamping and hence efficient time synchronization. Sometimes query-driven and event-driven schemes are used together [1], where the sink starts collecting data upon reception of various messages triggered by an interesting event on the sensor nodes.

We have however spotted several characteristics common to these categories of applications that could have an impact on the MAC layer. They are summarized in Table 1. Both categories share the fact that applications are tolerant to delays. In the former category, the collected data is analyzed once the experiment is completed. In the latter case, longer delays are preferred to packet loss. Even the surveillance application presented in [7] does not have much constraints on this matter.

In terms of size, only relatively small-scale sensor networks have been deployed, of the order of tens of sensors usually divided to smaller patches. The topology is always carefully chosen and the radio power tuned to guarantee the desired connectivity from startup. Most of the deployments are single-hop, sometimes a few hops are considered but hardly up to 6 hops. Though one exception has achieved 46 hops [3], it used a linear topology which significantly eased the routing and link management.

The length of deployments is usually less than a year, with most of WSNs being deployed from just a few days to a few weeks. This makes the energy management easier on the sensor node: cells in addition to solar panels are usually enough to sustain the energy demand during the experiment.

The interest in deployments of mobile WSN has recently increased [8, 9, 10], and mobility is foreseen as a very likely solution to expand the network coverage, improve the routing performance or the overall connectivity [11]. Of course, mobile sensors bring new challenges at the MAC layer (e.g. in terms of scheduling) but their induced constraints remained accessible so far. For example, the deployments depicted in [8, 9] only had respectively five and seven mobile sensors. Even though the experiment presented in [10] had hundreds of mobile devices, they were actually grouped into four patches in which single-hop communications occurred only few times a day. In every mobile scenario, the small number of nodes or the low frequency of communications only increased slightly the complexity of the deployments.

2.2 Simple Solutions to Simple Problems

While the research community keeps raising problems with the communication stack design (e.g. energy savings or data reliability), most of WSN deployments existing to date have managed to reach their goals. Some of them have lasted for several months without any particular intervention, meaning that both robustness and energy consumption of the deployed sensor nodes have been finely addressed. In addition, they were connected in a way that the data collection was pertinent and allowed scientists to perform interesting analysis following the deployment. For example, the glacier monitoring system deployed within SensorScope [6] lasted two months without any human intervention nor major breakdown. An analysis of the reported data also enabled the modelling of a particular micro-climate, thus allowing flood monitoring and prediction.

In order to understand how these deployments could have been successful, we investigated the communication stacks that have been used. We especially inspected the MAC layer, as its design has been of high interest for long among researchers of the WSN community. We could record a significant number of existing deployments relying on protocols built from scratch (e.g. in [2, 6, 7, 9]). The constraints induced by the application and the chosen hardware indeed governed a specific design of the MAC layer. As the current deployment characteristics result from basic constraints (i.e. small-scale or lowly dense networks, few hops, static sensors), scientists and engineers could design MAC layers with simple features while still matching the target application.

For instance, the people in charge of the SensorScope deployment equipped sensing stations with solar cells. The evaluation of the daily energy contribution allowed them to design a MAC layer without necessarily focusing on complex energy-saving mechanisms. Specifically, they came up with a high radio duty-cycle scheme (10% with all sensors being simultaneously active during 12 seconds every two minutes) that still ensured a successful long term experiment. In [2], the static and linear aspect of the network topology has led the authors to opt for a novel time-division scheme to control the medium access. This method indeed performed well provided that the network topology does not change.

As already mentioned, it resulted in successful deployments but it also contributed to making the reuse of these solutions difficult. In turn, these potential difficulties led to more and more MAC layers built from scratch. To our minds, this vicious circle raises two major drawbacks. First, it undoubtedly imposes WSN deployments to be performed by networking experts with strong programming skills in embedded systems. Second, the implemented solutions may become barely usable once confronted to slightly different deployment constraints. This will be further discussed in Section 5.

As brought up in [12], many problems remain open from the MAC layer perspective. This has led researchers to propose a vast range of MAC layer solutions dedicated to WSN. Before discussing why very few deployments are based on these solutions in Section 3.2, we first review the wide range of MAC protocols offered in the literature.

3. CONTROLLING MEDIUM ACCESS

3.1 Major Proposals

Research on protocols for Medium Access Control in WSN has been very prolific in the last decade. Especially, low power communication is the major challenge, in contrast with fairness or delay improvements in standard wireless networks. Overhearing, collisions, idle listening and control packet overhead have been identified as the main sources of energy wastage [13]. Tens of proposals have been put forward to address these issues, each promising great improvements in a specific area of the medium access algorithms.

Among the set of functionality provided by a MAC layer, scheduling has been the field of several enhancements in order to achieve drastic energy savings. The main idea is to put the radio in the sleep mode as much as possible while ensuring the link connectivity among the sensor nodes. Two major schemes initially came out in WSN in order to meet that challenge: sampling protocols and slotted protocols. More recently, hybrid protocols have also emerged to combine the benefits of both schemes.

Sensor nodes that employ sampling protocols always send a preamble before the plain data. Each node periodically wakes up and checks for such a preamble by sampling the channel. If no traffic is transiting over the air, the node goes back to sleep. If a preamble is detected, the node stays awake to receive the trailing data. The preamble length must indeed be longer than the sleep period on the node. This allows a loose synchronization of the sleep-listen period, though it increases the transmission cost on the sending node. B-MAC [14] and its Low Power Listening algorithm is one of the most famous sampling protocol for WSN.

Slotted protocols organize the nodes around a common schedule. Time is divided into slots distributed among the nodes, which agree with one another to use such slots to send or receive data, or to power off the radio. This scheme better distributes the transmission cost among all the nodes in the network, to the cost of time synchronization. Also, the nodes can hardly change their schedule without computing and distributing the timeslots again. Slotted CSMA schemes such as S-MAC [15] or IEEE 802.15.4 [16], as well as TDMA protocols (e.g. TRAMA [17] or DMAC [18]) belong to this category.

Most of the protocols in the literature are built on these schemes and propose enhancements targeting specific scenarios. MOBMAC [19] extends S-MAC with an adaptive frame size approach for mobile scenarios. DSMAC [20] also improves S-MAC with a dynamic duty-cycle mechanism to decrease the latency for delay-sensitive applications. Z-MAC [21] extends B-MAC with a hybrid CSMA/TDMA scheme to reduce collisions in networks with a medium to high contention level (e.g. in dense networks).

Likewise, other hybrid protocols have emerged and combine the concepts of the sampling and slotted schemes to get to best of each. Funneling-MAC [22] suggests the use of a CSMA scheme network-wide, while a TDMA algorithm is implemented around the sink in order to address bottlenecks in dense and multi-hop networks. SCP [23] combines scheduling and channel polling to reduce the preamble size used in sampling protocols. In mobile scenarios, MH-MAC [24] proposes to divide the frame time into a scheduled-based window for static nodes, and a contention-based window used by mobile nodes. The length of each window is dynamically adapted according to the ratio between mobile and static nodes.

Though this large diversity in the MAC protocols is supposed to satisfy a number of scenarios, very few deployments actually use MAC layers from the literature. Potential reasons are further explored in the next section.

3.2 From Theory to Practice

Among the MAC layers previously detailed in Section 3.1, only a few were employed during real deployments. To the best of our knowledge, their usage was limited to S-MAC in [25], B-MAC in [26, 27] and T-MAC in [28]. Besides, authors of [28] have discussed the difficulty to use solutions proposed in the literature. They also analyzed to what extent the failure of their deployment could be attributed to this design choice.

From a practical point of view, using MAC layer solutions from the literature has turned out to be complex. As exposed in the previous section, the MAC layers defined by the research community have addressed problems that have not been considered in existing deployments yet. Optimized power management, scalability or mobility are some of the major research topics in this area. However, the characteristics of the deployments are not that restrictive so far.

Moreover, the suggested MAC schemes propose very specific optimizations (e.g. on the backoff algorithm, or for the preamble size calculation) but do not define with precision the target application. This last point was already raised for the overall communication stack in [5], and the MAC layer is no exception. Translating the application needs into a set of required MAC optimizations is not an easy task, which could contribute to making the "built from scratch" solution more attractive.

Besides, the large majority of the research propositions are not implemented in the major operating systems dedicated to embedded sensors (such as TinyOS[1] or Contiki[2]). Their rough specifications on all the aspects beside the optimization they provide certainly prevent interested engineers or scientists to implement them.

Finally, apart from the MAC protocols proposed in the literature, an evident solution could be to use the standardized MAC protocol IEEE 802.15.4 [16]. Except for [1, 8], a survey of the radio chipsets embedded in most of the deployed sensors reveals that they do not belong to the category of hardware compliant with this standard. Though this limitation could be solved by the implementation of a software radio, the effort needed to cross it could still be greater than building a dedicated MAC layer from scratch. We can most likely assume that very few other deployments used it.

Overall, we could observe that a very small number of deployments have used MAC layers from the literature or even implementation code especially developed for previous applications. We now further investigate how to tend towards more convenient development methods.

4. AN APPARATUS FOR DEPLOYMENTS

Small-scale, single-hop, static and well-known topology are some of the characteristics of WSN deployments up to now (Table 1). As pointed in Section 2.2, the medium access can be tackled with simple protocols developed from scratch.

---

[1] http://www.tinyos.net

[2] http://www.sics.se/contiki/

In addition, hardware characteristics may also mandate specific code developments. This situation requires expert skills every new deployment. The preparation phase prior to a deployment could however be significantly alleviated by considering code reuse. Several schemes have been put forward to enable generality at the MAC layer while reducing the effort to implement a protocol on a specific hardware. We expose hereinafter the most interesting contributions that could be used if a WSN with similar characteristics as the already existing ones were to be deployed nowadays.

The need for a versatile protocol in WSN has led researchers to define B-MAC [14]. Besides its Low Power Listening mode mentioned in Section 3.1, tuning the MAC layer from the upper layers is another major contribution proposed by this protocol. A set of communication interfaces are specified to activate link-layer acknowledgments, configure the backoff time on a per-message basis, and adjust the preamble length and check interval of the Low Power Listening mode. One can then easily pre-configure B-MAC with the set of parameters suitable for the deployment characteristics. For example, reliability can be achieved by activating acknowledgments. Similarly, the desired throughput can be adjusted by modifying the Low Power Listening parameters. The authors of B-MAC have demonstrated that they could achieve deployments which could last about a year with over 98.8% packet delivery. Its flexibility makes it a likely solution for various scenarios. As a matter of fact, it is available in the TinyOS operating system for some of the wireless chipset drivers, and was already successfully used in some deployments [26, 27].

The MAC Layer Architecture (MLA [29]) greatly reduces the amount of code needed to implement a new MAC layer by proposing code reuse between different protocols and hardware platforms. The goal of MLA is twofold. First, it allows developers to easily implement a new MAC protocol by defining a component-based architecture. MLA divides the MAC layer into a set of reusable units. For example, a *preamble sender* component can be used in sampling protocols, while *time synchronization* is available for slotted protocols. By wiring these components together, a core can be built. Only the new code which makes the specificity of the protocol has to be developed. Next, by defining hardware-dependent and hardware independent components, the whole MAC layer can be ported to a different hardware with little effort. Only the hardware-dependent code has to be rewritten for the new platform. With this architecture, code reuse can be achieved up to 73%, while similar performance and memory footprint can be observed compared to monolithic implementations of the same protocols. MLA already implements five MAC protocols (including B-MAC) and is available on TinyOS.

In practice, these solutions still mandate a strong knowledge of the MAC layer. The use of B-MAC requires experimentation to find the correct set of configuration parameters. MLA needs to be extended with hardware or protocol-specific code. The time consumed to manipulate, configure and develop with such tools may refrain experts from using them, while developing a simple MAC layer would not necessarily take more time. However, as we will discuss in Section 5, the relative ease to design a simple communication stack every new deployment will be narrowed in the future. These tools thus definitely pave the way towards a unified solution for the foreseen deployments.

5. FUTURE NEEDS

5.1 Envisioned Shortcomings

With time, deployment constraints will undoubtedly raise new challenges when designing a suitable communication stack. Each of the characteristics presented in Table 1 will be applied on a broader scale: increase in the number of nodes, multi-hop topology, longer deployment, etc. In addition, both the topology of the network and the way it transports the information will evolve during the lifetime of the deployment. Higher mobility or dynamic change in the data collection scheme are two examples of such evolution.

In light of this, most of the MAC protocols designed for today's deployments would certainly face scalability issues if the same experiment was to be performed at a larger scale. First, they have not been designed to access the medium in dense networks. The static aspect of deployments makes them also very unlikely to behave efficiently in uncharted environments or where mobility would have a leading role (e.g. as depicted in [30]). As the collected data is not analyzed in real-time, short delays have not been considered yet. This may not be acceptable for responsive applications, such as target detection and tracking, that have hardly been considered in deployments so far. The rather short length of deployments or the use of solar panels does not give a great importance to power management. This has certainly kept engineers from optimizing the communication stack in term of energy consumption. For example, even though Sensorscope [6] was a successful long-term deployment, its 10% duty-cycle could be considered as excessive in WSN [31].

The research community is already acquainted with these aspects. For example, mobility and real-time constraints have been considered for a while as the future issues that need to be addressed at the MAC layer [12]. Still, this dynamic side of future deployments would require the MAC layer to perform well in various situations. For example, a WSN with multiple data collection schemes would first operate in an event-driven manner. Upon detection of a phenomenon, some nodes would start reporting the sensed data in a time-driven fashion for continuous monitoring purposes. As exposed in Section 2.1, both the time-driven and event-driven schemes have opposite requirements which could thus perform very differently on a single and monolithic MAC layer. The location of a given event is however unpredictable. It therefore excludes any prior installation of a specific MAC layer on nodes close to the spot of interest. Instead, the way to access the medium should be adjusted according to the constraints of the moment.

5.2 Ensuing Directions

As the complexity of the MAC protocol increases, specifying a new communication stack every new deployment will become a laborious task. This is especially relevant for scientists or engineers who do not have a wide knowledge in networking. It is crucial to think about robust protocols that could be suited to various deployment requirements as well as able to change their behaviour during the lifetime of the network. Keeping in mind the need for code reuse and versatility, we would like to investigate to what extent the solutions presented in Section 4 are appropriate to these foreseen deployments. Table 2 summarizes this discussion.

A versatile protocol such as B-MAC is well adapted to pre-configure the MAC layer of the sensor nodes prior to a

Identified Needs	*Short-term solutions*	*Shortcomings*	*Research directions*
Versatile protocols	B-MAC	Scalability	Protocols robust to the evolutions of the network
			Evaluation methods at large-scale
		On-the-fly reconfiguration	Translating context into suitable MAC parameters
			Predicting the impact on the network
Code reuse	MLA	Memory footprint	Dynamic component rewiring over the network

Table 2: Shortcomings and guideline for future researches

deployment. However, its operation in a dynamic scenario would require periodic re-computation of the adjustable parameters while the network is running. This is difficult to achieve as it raises two major considerations. First, translating a new context into what optimisations it requires at the MAC layer is far from being straightforward. A set of relevant input parameters must be identified and translated into a formalized description. For that purpose, an exhaustive list of which key characteristics could be used is suggested in [32], while a metadata format has been recently proposed in [33]. However, so far, no paradigm has ever been proposed to transcribe such parameters into configuration attributes made available to the MAC layer. Such computation algorithms remain a major and ambitious challenge: they must be designed to ensure a fully automated setup and composition of the medium access protocol. The second point to consider is the impact of such on-the-fly refinements on the whole communication stack and on the overall network. They require an extensive evaluation as an unfortunate modification could isolate a node, or worse, create a partition in the network.

Another shortcoming that we envision is more practical and is related to the cost of such flexibility in terms of both implementation effort and memory footprint. The use of an architecture such as MLA could considerably alleviate the development on sensor hardware. Even so, the implementation of the adaptive algorithms or components would increase the size of the MAC layer, which may no longer fit on memory constrained sensor nodes. Alternatively, dynamically rewiring the MLA components according to the needs could also be a likely solution. Though rewiring components on-the-fly is possible in today's WSN operating systems (e.g. with Contiki [34]), it still requires all the substitute components to be stored in memory. In both cases, the ability to build a minimal layer (for example at compilation time) and to retrieve the needed components while the network is running could tackle this problem.

Designing protocols that behave efficiently at large-scale while maintaining the coherence of the whole set of nodes will be a matter of prime interest. These protocols will have to succeed in both simulations and empirical evaluation at large-scale. For that purpose, wireless sensor testbeds such as moteLab[3] (190 motes available) will certainly get much credit from these new evaluation methods. To finely anticipate the impact of a versatile networking protocol, a key feature of these testbeds will also be their ability to emulate real environments. In this context, SensLab[4] (1024 open nodes expected with the ability to generate real radio noise and interference) will also contribute in the evaluation and experimentation of the envisioned algorithms and protocols.

6. CONCLUSION

Through a survey of WSN deployments, we have listed multiple common characteristics that could have an influence on the design of the MAC layer. Small-scale, single-hop and static deployments represent the vast majority of the WSN so far, and have been satisfied by simple MAC protocols governed by application, hardware and deployment constraints.

An analysis of the MAC layers proposed in the literature has revealed that most of the suggested solutions are not relevant with regard to the current needs. Furthermore, these proposals were hardly ever implemented in popular embedded systems or empirically evaluated. This has certainly limited the use of such solutions and strengthened the motivations to build protocols from scratch every new deployment.

With respect to the current deployments, such efforts could however be limited by considering code reuse. We have identified two contributions that could help in deploying new WSN with similar characteristics as the already existing ones. The B-MAC protocol can be adjusted prior to a deployment to suit the application requirements. Used in combination with MLA, implementing specific optimisations or porting the code to a different hardware can also be considerably simplified.

Still, the foreseen scenarios in WSN will consider large-scale and long-term deployments, while contemplating mobile and real-time applications. Such challenges will most certainly contribute in refraining people from designing their own communication stack from scratch. This will value solutions based on frameworks like B-MAC and MLA. However, the way the network evolves over time could also require protocols to dynamically adapt to the context. This will endorse the need for designing robust and more complex protocols.

To our minds, long term perspectives lie in the ability to design and configure a versatile MAC layer during the network lifetime. Anticipating the impact of these on-the-fly configuration changes on the network will also be a key challenge. As implementation and empirical evaluation of the solutions is an important step to ensure their robustness, some of these open questions for future researches will certainly find answers on the embedded operating system and architecture sides.

7. REFERENCES

[1] G. Werner-Allen, K. Lorincz, M. Welsh, O. Marcillo, J. Johnson, M. Ruiz, and J. Lees, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, pp. 18–25, March/April 2006.

[2] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat

---

[3] http://motelab.eecs.harvard.edu

[4] http://www.senslab.info

monitoring," in *WSNA '02: 1st International Workshop on Wireless Sensor Networks and Applications*, pp. 88–97, ACM, September 2002.

[3] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," in *IPSN '07: 6th Int. Conf. on Information Processing in Sensor Networks*, pp. 254–263, ACM, April 2007.

[4] I. F. Akyildiz et al., "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, pp. 102–114, August 2002.

[5] B. Raman and K. Chebrolu, "Censor networks: A critique of "sensor networks" from a systems perspective," *ACM SIGCOMM Computer Communication Review*, vol. 38, pp. 75–78, July 2008.

[6] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange, "Sensorscope: Out-of-the-box environmental monitoring," in *IPSN '08: 7th International Conference on Information Processing in Sensor Networks*, pp. 332–343, ACM/IEEE, April 2008.

[7] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh, "Energy-efficient surveillance system using wireless sensor networks," in *MobiSys '04: Int. Conf. on Mobile Systems, Applications, and Services*, pp. 270–283, ACM, June 2004.

[8] J. Allred, A. B. Hasan, S. Panichsakul, W. Pisano, P. Gray, J. Huang, R. Han, D. Lawrence, and K. Mohseni, "Sensorflock: an airborne wireless sensor network of micro-air vehicles," in *SenSys '07: 5th International Conference on Embedded Networked Sensor Systems*, pp. 117–129, ACM, November 2007.

[9] P. Zhang, C. M. Sadler, S. A. Lyon, and M. M. tonosi, "Hardware design experiences in zebranet," in *SenSys '04: 2nd International Conference on Embedded Networked Sensor Systems*, pp. 227–238, ACM, November 2004.

[10] B. Thorstensen, T. Syversen, T.-A. Bjørnvold, and T. Walseth, "Electronic shepherd – a low-cost, low-bandwidth, wireless network system," in *MobiSys '04: International Conference on Mobile Systems, Applications, and Services*, pp. 245–255, ACM, June 2004.

[11] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G. S. Sukhatme, "Robomote: Enabling mobility in sensor networks," in *IPSN '05: 4th International Symposium on Information Processing in Sensor Networks*, pp. 404–409, ACM, April 2005.

[12] M. Ali, U. Saif, A. Dunkels, T. Voigt, K. Römer, K. Langendoen, J. Polastre, and Z. A. Uzmi, "Medium access control issues in sensor networks," *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 33–36, April 2006.

[13] I. Demirkol et al., "Mac protocols for wireless sensor networks: A survey," *IEEE Communications Magazine*, vol. 44, pp. 115–121, April 2006.

[14] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *SenSys '04: 2nd International Conference on Embedded Networked Sensor Systems*, pp. 95–107, ACM, November 2004.

[15] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 12, pp. 493–506, June 2004.

[16] IEEE, "Part 15.4: Wireless medium access control (mac) and physical layer (phy) specifications for low-rate wireless personal area networks (wpans)," September 2006.

[17] V. Rajendran, K. Obraczka, and J. Garcia-Luna-Aceves, "Energy-efficient, collision-free medium access control for wireless sensor networks," in *SenSys '03: 1st International Conference on Embedded Networked Sensor Systems*, pp. 181–192, ACM, November 2003.

[18] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks," in *IPDPS '04: 18th International Parallel and Distributed Processing Symposium*, pp. 224–231, IEEE, April 2004.

[19] S. Ci, H. Sharif, and D. Peng, "An effective scheme for energy efficiency in mobile wireless sensor networks," in *ICC '04: International Conference on Communications*, vol. 6, pp. 3486–3490, IEEE, June 2004.

[20] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *WCNC '04: Wireless Communications and Networking Conference*, vol. 3, pp. 1534–1539, IEEE, March 2004.

[21] I. Rhee, A. Warrier, M. Aia, and J. Min, "Z-mac: a hybrid mac for wireless sensor networks," in *SenSys '05: 3rd International Conference on Embedded Networked Sensor Systems*, pp. 90–101, ACM, November 2005.

[22] G.-S. Ahn, S. G. Hong, E. Miluzzo, A. T. Campbell, and F. Cuomo, "Funneling-mac: a localized, sink-oriented mac for boosting fidelity in sensor networks," in *SenSys '06: 4th International Conference on Embedded Networked Sensor Systems*, pp. 293–306, ACM, October 2006.

[23] W. Ye, F. Silva, and J. Heidemann, "Ultra-low duty cycle mac with scheduled channel polling," in *SenSys '06: 4th International Conference on Embedded Networked Sensor Systems*, pp. 321–334, ACM, October 2006.

[24] A. Raja and X. Su, "A mobility adaptive hybrid protocol for wireless sensor networks," in *CCNC '08: 5th Consumer Communications and Networking Conference*, pp. 692–696, IEEE, January 2008.

[25] R. Cardell-Oliver, K. Smettem, M. Kranz, and K. Mayer, "Field testing a wireless sensor network for reactive environmental monitoring," in *ISSNIP '04: Intelligent Sensors, Sensor Networks and Information Processing Conference*, pp. 7–12, IEEE, December 2004.

[26] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong, "A macroscope in the redwoods," in *SenSys '05: 3rd International Conference on Embedded Networked Sensor Systems*, pp. 51–63, ACM, November 2005.

[27] R. Szewczyk, A. Mainwaring, J. Polastre, J. Anderson, and D. Culler, "An analysis of a large scale habitat monitoring application," in *SenSys '04: 2nd International Conference on Embedded Networked Sensor Systems*, pp. 214–226, ACM, November 2004.

[28] K. Langendoen, A. Baggio, and O. Visser, "Murphy loves potatoes: Experiences from a pilot sensor network deployment in precision agriculture," in *IPDPS '06: 20th International Parallel and Distributed Processing Symposium*, pp. 8–15, IEEE, April 2006.

[29] K. Klues, G. Hackmann, O. Chipara, and C. Lu, "A component based architecture for power-efficient media access control in wireless sensor networks," in *SenSys '07: 5rd International Conference on Embedded Networked Sensor Systems*, pp. 59–72, ACM, November 2007.

[30] V. Shnayder et al., "Sensor networks for medical care," in *Technical Report TR-08-05*, Division of Engineering and Applied Sciences, Harvard University, 2005.

[31] P. Dutta, D. Culler, and S. Shenker, "Procrastination might lead to a longer and more useful life," in *HotNets-VI: Sixth Workshop on Hot Topics in Networks*, ACM, November 2007.

[32] K. Römer and F. Mattern, "The design space of wireless sensor networks," *IEEE Wireless Communications*, vol. 11, pp. 54–61, December 2004.

[33] E. Meshkova, J. Riihijärvi, J. Ansari, K. Rerkrai, and P. Mähönen, "An extendible meta-data specification for component-oriented networks with applications to wsn configuration and optimization," in *PIMRC '08: Int. Symp. on Personal, Indoor and Mobile Radio Communications*, (Cannes, France), IEEE, September 2008.

[34] A. Dunkels, N. Finne, J. Eriksson, and T. Voigt, "Run-time dynamic linking for reprogramming wireless sensor networks," in *SenSys '06: 4th Int. Conf. on Embedded Networked Sensor Systems*, pp. 15–28, ACM, October 2006.