Bin-MAC: A Hybrid MAC for Ultra-Compact Wireless Sensor Nodes

Vahid Salmani and Pai H. Chou

Center for Embedded Computer Systems
University of California, Irvine
Irvine, CA 92697-2620, USA

{vsalmani, phchou}@uci.edu

CECS Technical Report 12-04
July 16, 2012

Bin-MAC: A Hybrid MAC for Ultra-Compact Wireless Sensor Nodes

Vahid Salmani
Center for Embedded Computer Systems
University of California, Irvine, CA, USA
vsalmani@uci.edu

Pai H. Chou
University of California, Irvine, CA, USA and
National Tsing Hua University, Taiwan
phchou@uci.edu

***Abstract*—This paper presents a light-weight hybrid protocol called Bin-MAC (Binary Medium Access Control) for highly resource-constrained wireless sensor nodes. In addition to simplicity and low footprint, a distinguishing feature of Bin-MAC is its deterministic contention resolution mechanism, which enables it to achieve bounded latency on data transmissions. As a result, Bin-MAC can be applied to delay-sensitive applications with real-time constraints, a feature not provide by most of the existing hybrid protocols. Another feature of the proposed protocol is that it requires carrier-sensing hardware only on the base station side, and sensor nodes do not have to possess such capability. Bin-MAC does not require clock synchronization, and thus clock drifts have no impact on its performance. Experimental results show Bin-MAC to be scalable and able to handle large network sizes without noticeable performance degradation.**

***Keywords*-hybrid protocol; ultra-compact nodes; collision detection; deterministic contention resolution;**

I. INTRODUCTION

In an *event-driven* WSN, sensor nodes transmit to the sink only when certain events occur (e.g., motion in case of infant monitoring) as opposed to continuous networks in which data is being reported at regular intervals. Event detection requires sensors to be vigilant most of the time. Passive vigilance, i.e., limited attention to ambient conditions, and adaptive sleep modes are techniques used to extend the lifetime of event-detection applications. The event-driven approach to sensor networks can support a wide range of applications from flood detection and human health status monitoring to target tracking.

Although many protocols have been proposed in the literature, few are actually widely used in real deployments, and most do not meet the current needs. There is still a need for dynamically adaptive protocols that adjust their operation according to the constraints of the moment [1]. The majority of adaptive protocols are not applicable to ultra-compact, low-complexity wireless sensor nodes for at least one of two important reasons. Either they require tight clock synchronization, which can be impractical, or they are too complex with a large footprint.

Some WSN applications require the sensor nodes to be very compact, such as monitoring of pre-term infants. Such nodes are highly constrained in terms of the radio range, processor speed, memory size, and power. In particular, limited memory implies a light-weight MAC protocol as well as a small data buffer [2]. Therefore, such applications require a bounded delay in transmitting data, or else part of the collected samples will be lost or overwritten.

This paper presents a hybrid MAC protocol for such ultra-compact wireless nodes used in event-driven wireless sensing applications. We concentrate on single-hop star-topology networks, as they are the most practical for our application and many of the real-world deployments [1].

II. RELATED WORK

A. *MAC Protocols*

Controlling access to a shared wireless medium may be based on contention, reservation, or hybrid approaches.

Carrier Sense Multiple Access (CSMA) is the most widely used contention-based protocol, in which nodes check for the absence of other traffic before transmission. It is popular due to its simplicity, flexibility, low overhead at low utilization, and obliviousness to global topology and clock synchrony. Nodes can dynamically join or leave without extra operations. However, the chief disadvantage of CSMA is collision, which causes energy waste. CSMA is efficient when the utilization is low, but the probability of collision increases rapidly as utilization increases. In general, contention-based MACs behave poorly under high contention and are not suitable for real-time or time-sensitive applications with stringent constraints.

Reservation-based schemes such as Time-Division Multiple Access (TDMA) divide time into slots. Each sensor node transmits only during its own time-slots. This approach is collision-free but requires synchronization, which may impose high overhead and complexity. Moreover, since sensor nodes do not usually have very accurate clocks, precise time synchronization is hard to achieve. Round Robin (RR) is the low-footprint form of the reservation-based approach, which does not require clock synchronization but it requires more control message exchange.

Hybrid schemes attempt to combine the best features of both approaches while offsetting their weaknesses. They try to adapt to different bandwidth conditions depending on demand. As an example, Z-MAC [3] is a hybrid method that runs CSMA under low contention and switches to TDMA under high contention. However, it does not guarantee bounded latency and thus may not be applicable to event-driven WSNs with stringent timing constraints.

Unfortunately, most of the existing protocols cannot be easily ported to highly resource-constrained sensor nodes such as Eco [4]. Reference [5] reports that a pure TDMA protocol designed for a single-hop star topology network requires 18-21 KB of code. This is five times the size of total EEPROM capacity of the Eco wireless sensing platform. According to the same reference, other approaches such as B-MAC, X-MAC and SCP use at least a memory footprint of 18 KB. Z-MAC is even more complex [6] and thus takes more memory.

Except for CSMA and RR, all of the mentioned protocols are impractical to implement on ultra-compact sensor nodes due to the relatively large memory footprint. Moreover, tight synchronization is difficult to achieve on low-complexity wireless motes [7]. In the presence of clock drift, most of the mentioned protocols will run into serious performance issues.

Although there is a tendency to make chips more compact, the size and cost will always be issues. Future trends for Systems-on-Chip (SoC) and Systems-in-Package (SiP) require packing the essential features while foregoing less important ones. For example, the nRF24L01 radio includes carrier sensing only on the stand-alone chip but not on the chip with the integrated micro-controller unit (MCU). As a result, low-complexity wireless sensor networks such as Eco require simpler and more efficient solutions. Furthermore, the application-centric work has shown that simple protocols, which obviate the unnecessary complexities, have worked and sufficed in practice [8].

B. Receiver-side Collision Detection

Collisions occurring at the receiver's end is one of the challenges in MAC design. It is generally believed that collisions prevent receivers from obtaining meaningful data and the energy drained in the transmission and reception of collided messages is just wasted [9]. Although most MAC protocols make the assumption that collisions between transmitters are solely destructive, it has recently been demonstrated that this is not always the case, and designers can take intuitive advantage of collisions [10]. Since collision detection is an important component of our work, we review different strategies for receiver-side collision detection (RCD).

The preamble-based approach exploits the *capture effect* to detect collisions [11]. This technique has been used for CC1000 (Mica2) radios. The success rate of the preamble-based collision detection drops quickly for more than two simultaneous senders.

The CRC-based approach [12] depends on checking the CRC bit of received messages and thus is applicable if the preamble and packet frame are received. Therefore, it cannot detect all types of collision.

The RSSI-based approach depends on frequently monitoring the RSSI information from the radio, but it entails additional processing burden on the MCU.

The carrier sensing-based approach is simpler and more reliable, and depends on sensing the medium for ongoing transmissions. It can detect collisions even when the preambles are not detected. On the CC2420 radio, this is done using the Clear Channel Assessment (CCA) signal. It is calculated by the radio chip and does not involve the MCU in collision detection [13].

C. Contention Resolution

Contention resolution is integral to contention-based and hybrid MAC protocols. The most common technique is back-off based collision avoidance as assumed by the popular IEEE 802.15.4. The problem with this approach is that it performs poorly under high contention. The RTS/CTS scheme alleviates the problem to some extent but it entails additional overhead. Variants of this scheme have been proposed to improve its efficiency using non-overlapping [14] and adaptive [15] contention windows.

Flip-MAC [10] employs a contention reduction technique with logarithmic complexity based on a series of probe-acknowledgment cycles. In each cycle, senders randomly set their ID to one of two possible addresses, and those who guessed correctly send simultaneous acknowledgment while the rest are out of competition. This process stops when a probe goes unacknowledged indicating that the contention level has dropped to a manageable level.

StrawMAN [16] handles contention based on the analogy of drawing straws. When a collision is detected, the simultaneous senders draw a random number for the length of the request signal, and the channel access is granted to the sender of the longest straw.

To the best of our knowledge, none of the existing contention resolution techniques is deterministic in terms of the waiting time prior to transmission. The reason is that they incorporate a random component that makes it impossible to guarantee a bounded delay. In the next section, we propose a deterministic technique to contention resolution, which is the key to Bin-MAC's adaptivity and bounded delay data transmissions.

III. DESIGN OF BIN-MAC

Bin-MAC works in *pull mode* (a.k.a. receiver initiated), meaning that the base station (BS) broadcasts query messages and the nodes reply to them. Since sensor nodes are assumed to be very resource constrained, we place most of the burden on the BS (a.k.a. sink). In the absence of a scheduling phase, Bin-MAC works in a round-robin style and is analogous to the *scheduled contention* scheme [7]. A distinguishing feature of this protocol is that the query message contains a range of node IDs instead of a single node ID. The proposed MAC is composed of four mechanisms,

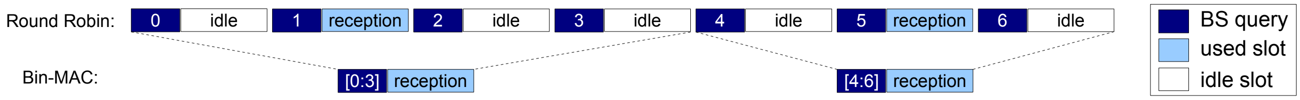

Figure 1: How Bin-MAC conserves bandwidth compared to Round Robin under low contention

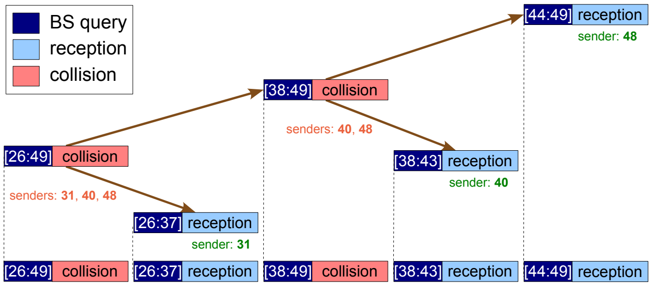

Figure 2: Example of Bin-MAC's BTCR operation

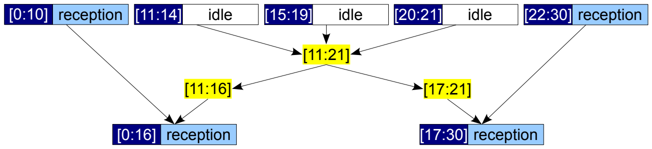

Figure 3: Example of Bin-MAC's slot consolidation

namely contention resolution, binary tree collision resolution (BTCR), slot consolidation, and duty cycle adjustment. We explain these mechanisms in the following subsections.

A. *Contention Resolution*

Although Bin-MAC is a reservation-based protocol, it provides a mechanism for handling contention, which accounts for its hybrid behavior. This mechanism is performed differently from contention-based MACs in which nodes have carrier-sensing capability and can sense the channel before transmission. We allow a node to transmit in certain time slots and acquire its own time slot in a deterministic manner as follows.

The BS, which has the knowledge of overall network condition, is responsible for contention resolution. We assign each time slot to a range of nodes e.g., [0:3]. A slot, most of the time, has one *active* transmitter; however, all the nodes whose ID is within the range (in this example between 0 and 3 inclusively) can transmit in the slot. The active transmitter of a slot is the node that has already been sending data to the sink in that slot. Other nodes in the range have recently been inactive, but in case they detect an event they will have the chance to use this slot to inform the BS that they have detected an event and want to acquire their own slot.

Fig. 1 shows how Bin-MAC conserves the unused slots compared to RR. If the active transmitter and some already inactive nodes send in the same slot, a collision occurs, and the BTCR mechanism described in the next subsection (III-B) is used to resolve the collision and assign new slots to the recently activated nodes.

B. *Binary Tree Collision Resolution*

In response to the BS's pull message, three cases may happen. (1) A data message is successfully received from one of the nodes in the range. (2) There is no response, so the slot is unused and will be treated as described in the next subsection. (3) A collision is detected *at the BS* using any of the RCD techniques discussed in Section II-B, which will be resolved as follows.

To enable receiver-side collision detection, nodes *broadcast* the data message [13]. When the BS detects a collision, it infers that two or more nodes in the range specified in the pull message have transmitted simultaneously. Therefore, it splits the range in half and issues a new pull command for each half. This will continue until the collision is resolved.

This process conceptually creates a *"binary tree"* whose leaves are the slots and apparently the slots having the smallest possible range will contain a single node ID. We take a *branch and bound* approach in the sense that we stop branching once we find the corresponding slot of a branch to be idle. Fig. 2 shows how Bin-MAC resolves a collision between nodes 31, 40, and 48 that belong to the range [26:49] and transmit in the same slot.

C. *Slot Consolidation*

If the BS neither receives a message nor detects a collision in response to its query message, it infers that all the nodes in the range are inactive, and that no node is using the corresponding slot. Since idle slots waste the bandwidth, the BS saves such slots by removing them and combining their range with those of the adjacent non-idle slots. Note that the BS may keep a counter for each slot to remove only those slots that have been idle for a number of consecutive rounds. This would be helpful in presence of a bad link to handle temporary connection losses.

At the end of each round the BS merges the ranges of all the adjacent idle slots into one, splits it into half and merge each half with the adjacent non-idle slot's range. If the idle slot is the first or the last, it will not be split. Instead, its whole range is combined with its only adjacent slot. Fig. 3 shows how the slot consolidation procedure helps Bin-MAC save bandwidth.

D. *Duty Cycle Adjustment*

Bin-MAC works in the round-robin pull mode, so it is essential to conserve energy by keeping the nodes in sleep

Table I: Possible cases for a time-slot

slot status	BS reaction	slot count
reception	querying as before	unchanged
idle	removing the slot	decremented
collision	resolving the collision	incremented

mode until they are really being pulled. For that purpose, nodes need to know the number of slots in the round. Since the number of slots can change dynamically, each pull message contains the current number of slots as well.

At the beginning of each round, the BS knows the number of slots. As the BS queries the nodes, it updates the number of slots based on what happens in the current slot. If the slot is idle, it will be removed in the next round and thus, the BS decrements the number of slots. In case of a collision, the number of slots is incremented because the slot will be split into the two halves.

Upon receiving the pull message and based on the number of slots, a node can determine if it can save energy by going to sleep mode until it is being queried in the next round. A node enters sleep mode only after a successful transmission, which can be inferred from an ACK from the BS, or examining the next pull message in case packets are not acknowledged. If the next message's range does not contain the node's ID, the transmission has been successful. Otherwise, a collision has occurred. Note that a node might have to wait for a while after waking up because of an increase in the number of slots, but it will never miss its slot due to a late wakeup. Table I summarizes the three possible states for a time-slot and the decision made by the BS in response to each one.

E. Protocol Operation

Bin-MAC needs to be provided with a range of the operating node IDs in the network. The range is specified using a pair of minimum and maximum IDs, i.e., $[min : max]$, but these IDs do not have to actually exist in the network. The only constraint is that all the existing nodes must have an ID within the specified range. Therefore, these IDs should be selected loosely enough so that the range can address an arbitrary number of nodes that might join the network at any point during the network operation.

Assuming that the specified range is $[0 : n]$, the BS periodically broadcasts a query message containing this range. As long as there is zero or one sender among nodes, the BS keeps sending the query message containing this same range. We keep sending the query even if there is no active node in the network so that in case of detecting an event, the corresponding node can transfer its data as soon as possible, thereby minimizing transmission delays. Since the BS is typically line powered, its energy usage is less of a concern. If two or more nodes reply to the same query, the BS performs the BTCR mechanism described in Section III-B.

Any node that detects an event will wake up and listen to the BS until it receives a query message containing a range including its ID. In response to such a message, the node transmits a data message. If the message arrives at the BS successfully, the BS may reply with an ACK or moves on to the next slot depending on protocol configuration. In case of a successful transmission and based on the number of slots, the node may go into sleep mode to save energy. In case of a failed transmission, the node keeps listening to the BS to participate in the BTCR procedure.

At the end of each round, the BS removes the idle slots and merge their ranges with the active slots, as explained in Section III-C.

IV. DISCUSSION

In this section we discuss and analyze the worst-case performance of the proposed protocol because it is of particular concern to real-time system designers. We also compare and contrast the features of Bin-MAC with those of some related work from various aspects.

A. Worst Case Analysis

The worst case of BTCR occurs when two nodes having subsequent IDs both detect an event and thus have to acquire their own time slot. Since these nodes are two sibling leaves in the conceptual binary tree, collision will take $O(\lg n)$ to be resolved. This situation could be avoided to a great extent by having the nodes scattered randomly in the intended area of operation. Since event-driven networks usually have spatially-correlated contention, nodes that are close to the event of interest will be active simultaneously. If adjacent nodes do not have subsequent IDs, spatially-correlated contention will not lead to the worst-case collision resolution.

The worst case in the network operation happens when all nodes switch to active state simultaneously. This is a rare case, but we analyze it to show the competence of Bin-MAC. In such a case, the BS would take $2n$ query messages to resolve all the collisions and assign time slots to all nodes because it would conceptually create a complete binary tree. This case, which lasts only one round, would still be $O(n)$, which is the same as that of RR. If half of the nodes detect an event at the same time, which is still a rare case, then the BS broadcasts n queries, which is equivalent to the length of a round in RR. The best case for Bin-MAC is $O(1)$ while it is still $O(n)$ for RR.

The worst case of slot consolidation occurs when all nodes stop sending simultaneously. The BS will query all the nodes and find that all the slots are idle, in which case it takes $O(n)$ for the BS to combine all the slots into one.

Since the BTCR mechanism takes $O(\lg n)$ and a node has to wait $O(n)$ for its turn to use the channel, the node can transmit its packet in $O(n)$ at the latest. This is how Bin-MAC provides a measurable, bounded transmission latency.

Table II: Comparison of CSMA, Bin-MAC, RR, and Z-MAC

		CSMA	Bin-MAC	RR	Z-MAC
Prior knowledge of node IDs		No	No	Yes	Yes
Bounded latency guaranteed		No	Yes	Yes	No
Carrier sensing on nodes		Yes	No	No	Yes
Cost of dynamic join/leave		None	None	High	Low
Channel Utilization	Low contention	High	High	Low	High
	High contention	Low	High	High	High
Fairness		Poor	Good	Good	Good
Memory footprint		Low	Low	Low	High
Multi-hop operation		Yes	No	No	Yes
Clock synchronization		No	No	No	Yes

The worst case for the node operation is to alternate between sleep and active mode in each round. Although this is unlikely to happen in an event-driven application, a similar condition may take place with a noisy or bad connection. To handle this case, the BS may simply keep a counter for each slot to remove only those slots that have been idle for a number of consecutive rounds.

B. *Features*

Bin-MAC is highly flexible. In the case that there is only one active node in the network, the protocol can operate using a single slot, thereby minimizing the latency. As the number of active nodes increases, the number of time slots is proportionally increased. In saturation conditions, every node is given its own time slot and thus the protocol operates like the RR protocol, making the full use of the bandwidth.

Bin-MAC is one of the simplest hybrid protocols ever designed and has a very small memory footprint. It adds no complexity to the sensor nodes compared to RR. On the Eco platform, RR has a memory footprint of around 2KB. Therefore, Bin-MAC would take up around the same space if implemented on an Eco sensor node. On the other hand, the proposed MAC does add complexity to the BS for managing the slots. However, the BS's complexity is usually not a concern as it is assumed to be a powerful device. With Bin-MAC, nodes do not require carrier sensing capability and it is only a requirement for the BS. This is an advantage, because this feature is likely to be omitted in integrated radio-MCU SoCs (Section II-A).

Unlike other hybrid protocols, Bin-MAC provides bounded latency, which is an important requirement of wireless sensing applications with stringent real-time constraints. This feature of Bin-MAC has to do with the fact that, contrary to other hybrid MACs, it does not rely on any random procedure to resolve the contention. With an ordinary hybrid MAC, a packet transmission may theoretically fail any number of times due to a collision, and there is no limit on the occurrence of two or more nodes sensing the channel at the same time, finding it silent and thus transmitting simultaneously. Nonetheless, Bin-MAC is a *greedy* approach that intuitively makes use of every collision to progressively resolve the contention.

Table III: TI/Chipcon CC2420 radio model parameters

Parameter	Value
Bit rate	250 kbps
RX/TX switching delay	192 µs
CCA sampling time	128 µs
Radio on/off transition time	1.792 ms

As a hybrid protocol, Bin-MAC adapts to the level of contention. It combines the strength of contention-based and reservation-based approaches while offsetting their weaknesses. Similar to CSMA/CA, it achieves high channel utilization and low latency under low contention. It also does not require prior knowledge of existing node IDs in the network. The only requirement is a rough estimate of minimum and maximum IDs of those nodes that either are present in the network or might join the network in the future. These limits are easy enough to determine loosely such that the IDs of nodes that dynamically join the network fall within the covered range. Obviously, nodes may leave the network dynamically without any cost.

On the other hand, Bin-MAC achieves high bandwidth utilization and fairness under high contention similar to RR. As long as the link quality is good, it guarantees bounded and measurable latency even under high contention. Table II compares the mentioned light-weight protocols as well as Z-MAC from various aspects.

V. PERFORMANCE EVALUATION

We have simulated the Telos [17] platform, which incorporates a TI/Chipcon CC2420 radio transceiver. Telos supports carrier sensing and the timings of CC2420 is widely studied and well understood. Table III shows the timer values taken from the datasheet. We have also taken the processing times into account using the measurements presented in [18]. Most notably, the transfer time from MAC layer's data FIFO buffer into the buffer of the CC2420 transceiver is determined by the following equation:

$$\text{TX buffering time} = 0.31\text{ms} + (x + 11).46\mu\text{s} \qquad (1)$$

where x denotes the payload length in bytes. Analogously, the transfer time from the CC2420 buffer to the MAC layer's data buffer is represented by:

$$\text{RX buffering time} = 1.28\text{ms} + (x + 5).46\mu\text{s} \qquad (2)$$

We have also implemented the automatic acknowledgment feature of the CC2420 radio transceiver.

To evaluate Bin-MAC's performance, we compare it with RR and CSMA/CA, which do not require clock synchronization and are light-weight enough to be implemented on ultra-compact sensor nodes such as Eco. We implemented CSMA/CA according to the IEEE 802.15.4 standard.

Additionally, we used the ideas behind Z-MAC and implemented a simplified version of it for a single-hop, star topology network. We call this protocol Z-MAC* and

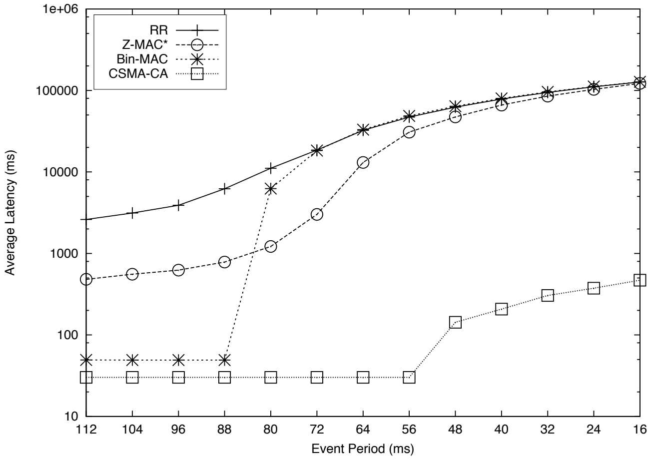

Figure 4: Average latency versus event period

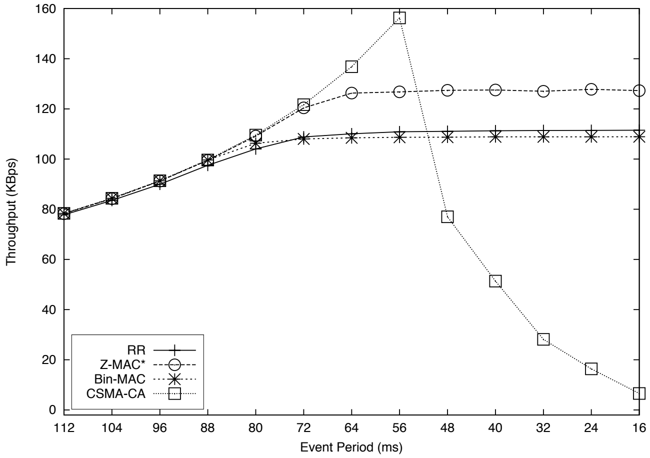

Figure 5: Throughput versus event period

assume there is no clock drift on sensors, which is the best case for Z-MAC*.

The network is star topology composed of 20 sensor nodes and a base station. Simulation time is 2x10^7 symbol periods. We vary event period from 112 ms to 16 ms to evaluate the performance under both low and high contention. Each event is assumed to require 10 successful transmissions (1100 bytes) to be reported completely. We assumed data buffer on sensor nodes to be unlimited to enforce successful delivery of events. The algorithms are compared in terms of latency, throughput, success rate, energy efficiency, and scalability.

A. *Latency*

Fig. 4 shows the average latency as a function of event period. Under low contention, Bin-MAC performs very closely to CSMA/CA. Z-MAC* shows a higher latency because of the fact that in most cases nodes are not the owner of the current slot and therefore need to wait before sensing the channel. Obviously, RR has the highest average latency. As the contention increases, all protocols except CSMA/CA show sudden growth in latency. The reason for CSMA/CA's lower latency under high contention is that it imposes a limit on the number of times it tries sending a packet, while other protocols keep sending until it is finally delivered. At some point, Z-MAC* shows better performance than Bin-MAC due to the fact that Bin-MAC's contention resolution introduces extra latency, but they converge in performance as contention increases.

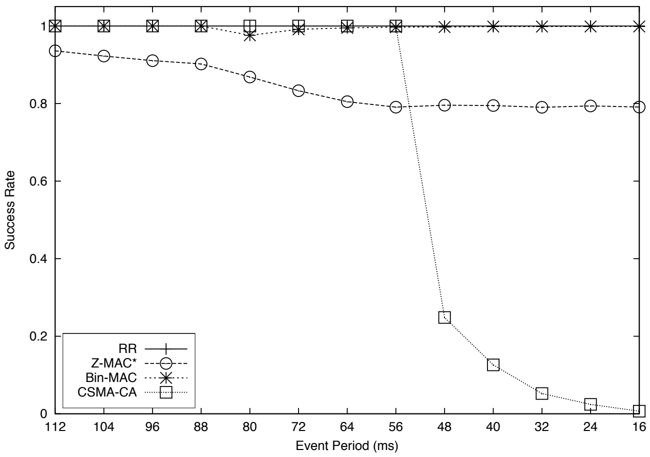

Figure 6: Success rate versus event period

B. *Throughput*

Fig. 5 shows the average throughput. Under low loads, Bin-MAC utilizes the bandwidth as much as possible. As the system's load increases, Bin-MAC performs closely to RR. Z-MAC* demonstrates a higher throughput due to lower control packet overhead. However, as we show later, Bin-MAC outperforms Z-MAC* in terms of throughput as network size grows.

C. *Success Rate*

Success rate is defined as the ratio of successfully transmitted data messages to the total number of data transmissions. It can be indicative of the impact of collisions on protocol performance. Fig. 6 depicts the observed success rate as a function of event period. As the contention goes up, the performance of Z-MAC* and CSMA/CA drops while Bin-MAC performs closely to RR whose success rate is 1. This means collisions have a negligible impact on Bin-MAC's performance.

D. *Energy Efficiency*

In the context of WSNs, communication-related energy consumption has a direct relation to *duty cycle*. Duty cycle is defined as $\Delta_{\text{on}}/(\Delta_{\text{on}}+\Delta_{\text{off}})$ where Δ_{on} is the radio on time and Δ_{off} the radio off time. Δ_{on} includes the time the radio is sending, receiving, or listening. These three modes of operation require roughly the same energy [18].

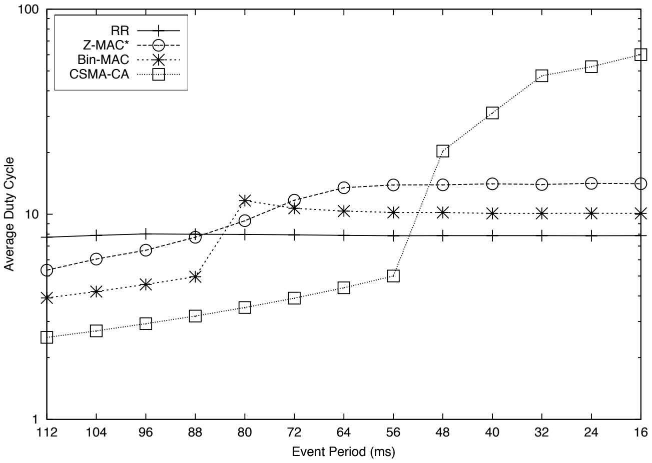

Figure 7: Average duty cycle versus event period

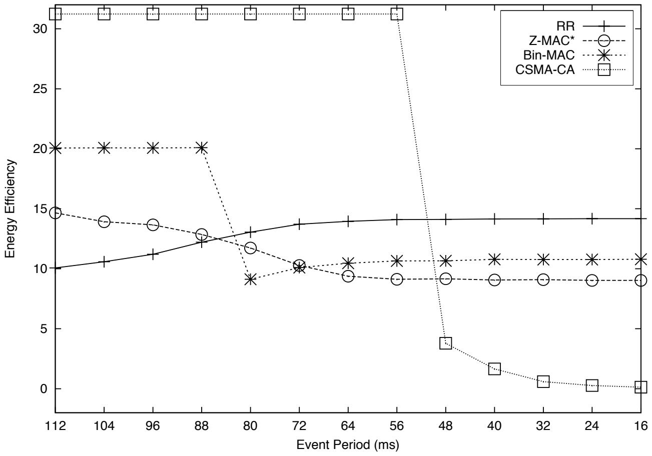

Figure 8: Energy efficiency versus event period

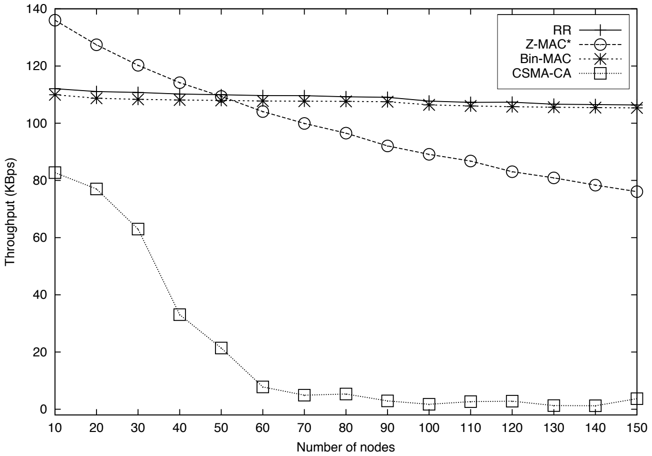

Figure 9: Throughput versus network size

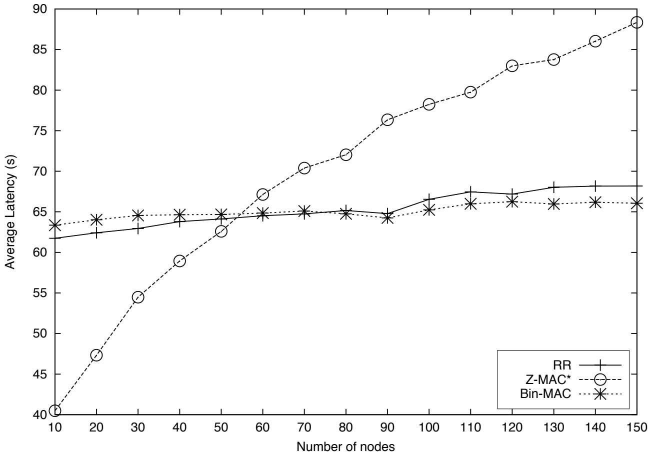

Figure 10: Latency versus network size

Fig. 7 shows the average duty cycle. Under low contention, Bin-MAC shows very close performance to CSMA, and under high contention, it performs closely to RR. Bin-MAC also shows better performance than Z-MAC* in most cases.

We define energy efficiency in terms of throughput over average duty cycle and depict it in Fig. 8. Under low contention, Bin-MAC is very energy-efficient, and in most cases it is more energy efficient than Z-MAC*.

E. Scalability

Scalability has been a challenge in MAC design [1]. Reference [9] reports that no MAC as of today is proven to be highly scalable, and that current MACs have trouble handling a large number of nodes.

To evaluate the protocols from the scalability aspect, we vary the number of nodes from 10 to 150 under the same overall load. The event period is fixed at 48 ms, or medium contention level.

Fig. 9 shows the achieved throughput as a function of network size. Bin-MAC and RR show good scalability, while CSMA and Z-MAC* show performance degradation as the number of nodes increases. This result confirms the fact that contention-based collision-avoidance protocols have scalability problem [19].

Fig. 10 depicts the latency as a function of the number of nodes. Obviously, CSMA has the lowest latency, but we omitted its curve to better show the performance of the other protocols. Bin-MAC and RR show some increase in latency but they are still scalable, while Z-MAC* shows considerable growth in latency as the network size grows. Our results suggest that the proposed binary algorithm is well-suited for handling larger scales.

VI. CONCLUSIONS

We propose a very light-weight hybrid MAC protocol called Bin-MAC for ultra-compact wireless sensor nodes in a star topology network. We take a deterministic, binary tree-like approach to contention resolution, which makes

Bin-MAC suited for applications with stringent real-time constraints. Our protocol does not require clock synchronization, nor does it require carrier sensing hardware on sensor nodes. We show that Bin-MAC can adapt to the contention level very well, and that it can handle larger scales very well. We plan to apply Bin-MAC to wireless monitoring of infants in incubators. For this purpose, we will implement it on a real-world platform such as Eco, and conduct more comprehensive performance evaluation.

ACKNOWLEDGMENT

This work is sponsored in part by National Science Foundation (NSF) grant CBET-0933694, UCI Technology Development Fund (TDF), Grant UL1 RR031985 from National Center for Research Resources (NCRR) as a component of the National Institute of Health (NIH) and the NIH Roadmap for Research. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF. The authors also acknowledge support and collaboration by Dan Cooper, Dana Gravem, and Julia Rich of the Institute for Clinical Translational Science (ICTS), and Donald Patterson of the Department of Informatics, UC Irvine.

REFERENCES

[1] R. Kuntz, A. Gallais, and T. Noel, "Medium access control facing the reality of WSN deployments," *ACM SIGCOMM Computer Communication Review*, vol. 39, pp. 22–27, June 2009.

[2] M. Patel and J. Wang, "Applications, challenges, and prospective in emerging body area networking technologies," *IEEE Wireless Communications*, vol. 17, no. 1, pp. 80–88, 2010.

[3] I. Rhee, A. Warrier, M. Aia, J. Min, and M. L. Sichitiu, "Z-MAC: a hybrid MAC for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 511–524, 2008.

[4] "Ecomote," http://www.ecomote.net/.

[5] K. Klues, G. Hackmann, O. Chipara, and C. Lu, "A component-based architecture for power-efficient media access control in wireless sensor networks," in *Proceedings of the 5th ACM International Conference on Embedded Networked Sensor Systems (SenSys)*. Sydney, Australia: ACM, November 2007, pp. 59–72.

[6] S.-C. Kim and C.-W. Woo, "A scalability analysis of TDMA-based ad hoc MAC protocols," in *Proceedings of the International Conference on Computational Science and Its Applications (ICCSA)*. Springer-Verlag, 2009, pp. 580–592.

[7] V. Salmani and P. H. Chou, "Pushing the throughput limit of low-complexity wireless embedded sensing systems," in *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC)*. IEEE Computer Society, 2010, pp. 181–188.

[8] B. Raman and K. Chebrolu, "Censor networks: a critique of "sensor networks" from a systems perspective," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 3, pp. 75–78, July 2008.

[9] T. W. Abdelmalik Bachir, Mischa Dohler and K. K. Leung, "MAC essentials for wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 12, pp. 222–248, April 2010.

[10] D. Carlson and A. Terzis, "Flip-MAC: A density-adaptive contention-reduction protocol for efficient any-to-one communication," in *Proceedings of the 7th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE, 2011, pp. 1–8.

[11] K. Whitehouse, A. Woo, F. Jiang, J. Polastre, and D. Culler, "Exploiting the capture effect for collision detection and recovery," in *Proceedings of the 2nd IEEE workshop on Embedded Networked Sensors (EmNetS-II)*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 45–52.

[12] G. Chockler, M. Demirbas, S. Gilbert, and C. Newport, "A middleware framework for robust applications in wireless ad hoc networks," in *Proceeding of the 43rd Allerton Conference on Communication, Control, and Computing*, 2005.

[13] M. Demirbas, O. Soysal, and M. Hussain, "A singlehop collaborative feedback primitive for wireless sensor networks," in *In Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2008, pp. 126–130.

[14] Q. Tian and E. J. Coyle, "A MAC-layer retransmission algorithm designed for the physical-layer characteristics of clustered sensor networks," *IEEE Transactions on Wireless Communications*, vol. 5, no. 11, pp. 3153–3164, 2006.

[15] A. Haque, M. Murshed, and M. Ali, "Efficient contention resolution in MAC protocol for periodic data collection in WSNs," in *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference (IWCMC)*. ACM, 2010, pp. 437–441.

[16] F. Österlind, N. Wirström, N. Tsiftes, N. Finne, T. Voigt, and A. Dunkels, "StrawMAN: making sudden traffic surges graceful in low-power wireless networks," in *Proceedings of the 6th Workshop on Hot Topics in Embedded Networked Sensors (HotEmNets)*. ACM, 2010, pp. 14:1–14:5.

[17] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN)*. IEEE Press, 2005.

[18] P. Suriyachai, U. Roedig, and A. Scott, "Implementation of a MAC protocol for QoS support in wireless sensor networks," in *Proceedings of the 2009 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. Washington, DC, USA: IEEE Computer Society, 2009, pp. 1–6.

[19] Y. Wang and J. J. Garcia-Luna-Aceves, "Collision avoidance in multi-hop ad hoc networks," in *Proceedings of the 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS)*. IEEE Computer Society, 2002, pp. 145–154.