CBA-EVT: A Traffic-Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks

Te-Li Wang, Jung-Chun Kao, and Si An Ciou
Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan

***Abstract*—For lifetime extension, energy-efficient *m*edium-*a*ccess *c*ontrol (MAC) protocols designed for battery-powered wireless sensor networks (WSNs) usually enable nodes to operate in low duty cycle. However, without offline change in system parameters, existing protocols either save energy at a cost of long latency or work well only for a narrow range of traffic load. To remedy these problems, this paper presents CBA-EVT, an energy-efficient MAC protocol that is self-organizing in the presence of different operating regimes. To be self-organizing, CBA-EVT leverages two techniques—*c*ost *b*enefit *a*nalysis and *e*xtreme-*v*alue *t*heory. CBA-EVT uses the cost benefit analysis that considers both delay and energy consumption to determine the sleep schedule from a macroscopic view; while it exploits extreme value theory to adjust duty cycle at a microscopic aspect. Through simulation, we confirm that compared with existing energy-efficient MAC protocols designed for WSNs given a delay bound, CBA-EVT with a single set of system parameters achieves significant energy savings at a wide range of traffic load.**

***Keywords*—energy efficiency, medium access control, wireless sensor network**

I. INTRODUCTION

A wireless sensor network (WSN) typically consists of a large number of battery-powered sensor nodes. For lifetime extension, it is of utmost importance in WSNs to design an energy-efficient medium-access control (MAC) protocol that minimizes energy consumption while achieving the end-to-end delay constraint to meet applications' requirements. Duty cycling, which switches nodes between awake and sleep states, is a critical technique for energy savings. However, without offline change in system parameters, existing protocols either save energy at a cost of long latency (particularly at a medium or high traffic load) or work well only for a narrow range of traffic load. They are not self-organizing in the presence of different operating regimes.

For example, S-MAC [3] in which nodes have fixed duty cycle lacks the ability to adjust node's duty cycle at run time. Many other protocols have the ability of duty cycle adjustment. However, their ability is very limited for the scenario of having a wide range of traffic load.

To remedy the aforementioned problems, this paper proposes *CBA-EVT*, an energy-efficient MAC protocol in WSNs that is highly self-organizing. CBA-EVT is the first energy-efficient MAC protocol in WSNs that exploits *c*ost *b*enefit *a*nalysis and *e*xtreme *v*alue *t*heory, aiming to be highly reactive and adaptive to a wide range of online network traffic. We develop several mechanisms in order to minimize the dependency on system parameters, which in general can be optimized only for a single operating point or a narrow range. To save energy but not cause long delay in the presence of different operating regimes, CBA-EVT takes both energy consumption and delay into account.

CBA-EVT is a receiver-initiated approach: Each sensor node schedules when to wake up for packet reception; a sender wakes up in light of receiver's schedule and then goes to sleep after transmitting all packets to the receiver. More precisely, each sensor node schedules its own reception slots on a per-frame basis and adjust its own duty cycle on a per-slot basis. A time frame consists of a number of time slots; reception slots are the time slots in which a sensor node wakes up to receive packets. Beacons are broadcast to notifies one-hop neighbors of the reception slots. Unlike sender-initiated protocols which reduce duty cycle only at receivers, receiver-initiated protocols reduce duty cycle at both senders and receivers [6].

To be self-organizing and adaptive to a wide range of time-varying traffic load, CBA-EVT has a mechanism of online traffic estimation. The estimated energy consumption and packet delay are transformed into cost and benefit quantities. Then CBA-EVT uses the cost-benefit analysis to decide the schedule of reception slots.

For further energy savings, a sensor node does not have to remain awake all the time in a reception slot; instead it can go to sleep at the *completion time* (because it is not involved in events in the remaining reception slot). In reality, completion time is not deterministic because there are multiple nodes competing for access to a shared wireless channel in a random access manner. So we apply extreme-value theory to predict completion time for each slot. After the predicted completion time, a sensor node will turn itself off if no packet is received for a predefined time interval.

The rest of this paper is organized as follows. In Section II we present related work. We present the overview and details of CBA-EVT in Section III. Section IV presents a detailed evaluation of our approach using simulation. Finally, we present some concluding remarks in Section V.

II. RELATED WORK

Among energy-efficient MAC protocols designed for WSNs, S-MAC [2],[3] is one of the pioneers. To save energy, S-MAC proposes duty cycling. Sensor nodes sleep and wake

---

This research was supported in part by National Science Council, Taiwan, under grant no. NSC 102-2221-E-007-011.

up periodically with a fixed duty cycle. Duty cycling effectively reduces energy wasted on idle listening, which is the dominant source of waste of energy in WSNs.

T-MAC [4] relaxes the fixed duty cycle requirement imposed on S-MAC. Whereas all nodes under S-MAC (in the same time domain) wake up and fall into sleep simultaneously, nodes under T-MAC wake up simultaneously but fall into sleep at different times. Under T-MAC, an active period at a node ends when carrier sensing finds the channel idle for a given time duration.

Similar to T-MAC, DSMAC [1] and AMAC [5] also relax the fixed duty cycle requirement imposed on S-MAC. The major difference of their mechanisms from T-MAC is that DSMAC and AMAC adjust duty cycle of a node by halving or doubling the wake-up interval. Halving (doubling) the wake-up interval does not change the length of listen time in a wake-up interval but it shortens (lengthens) the sleep time. Besides, AMAC carefully redesigns the packet format and introduces a special SYNC packet, which enables to AMAC have the listen period shorter than S-MAC and DSMAC.

Unlike the aforementioned protocols which are all sender-initiated. PW-MAC [6] is receiver-initiated. Sender-initiated protocols reduce duty cycle at only receivers, whereas receiver-initiated protocols reduce duty cycle at both senders and receivers. In PW-MAC, nodes themselves decide when to wake up by using a pseudo-random number generator. The wake-up information is broadcast through beacon packets upon each wake-up and potential sender nodes will wake up before receiver's next wake-up for packet transmission to the receiver. PW-MAC achieves near-optimal energy efficiency both at receivers and at senders [6], when the traffic load is low and system parameters are appropriately chosen offline. One disadvantage of PW-MAC is the lack of the ability to adjust wake-up intervals according to dynamic network condition. PW-MAC's energy efficiency could degrade in the presence of different operating regimes.

III. CBA-EVT DESIGN

We first define the terminology used in this paper. The contention-based wireless channel is divided into time *frames*. A time frame is further divided into L time *slots*[1]. As shown in Fig. 1, a node wakes up at the beginnings of all or a part of slots for packet transmission (in *transmission slots*), packet reception (in *reception slots*), or both (in *hybrid slots*).

For exposition purpose, the spacing between consecutive frames in Fig. 1 is beacon periods in which beacon packets are broadcast to one-hop neighbors for wake-up advertisement. Beacons are not required to be sent in exclusive time intervals. Indeed, beacons can also be sent inside any slot in a frame.

CBA-EVT is receiver-initiated: Each node decides its own schedule of reception slots; a sender learns receiver's reception slots via beacons and sets its transmission slots to be receiver's reception slots. To save energy, an awake node does not have to remain awake in the entire time slot. After finishing its jobs at *completion times*, a node could enter the sleep mode for energy savings. The following subsections explain the two main functions of CBA-EVT—reception slot scheduling (on a per-frame basis) and duty cycle adjustment (on a per-slot basis)—that are both designed to adapt to online network traffic.

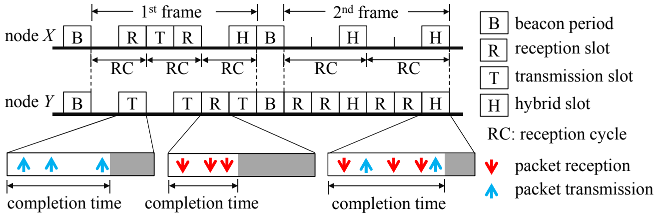

Fig. 1. System overview of CBA-EVT. The frame size here is 6 slots.

A. *Dynamic Per-Frame Reception Slot Scheduling*

In WSNs, there is a tradeoff between the two conflicting objectives—energy consumption and packet delay minimizations. The more reception slots in a frame, the shorter packet delay but the larger energy consumption. In CBA-EVT, each node deals with this multi-objective optimization problem by using the cost benefit analysis (CBA) on a per-frame basis; the result is the reception slot schedule. To determine the schedule according to online network traffic, an online estimate of the number of packets that will be received in the coming frame is fed into the cost benefit analysis.

1) Online Traffic Estimation

At the beginning of a frame, each node updates the estimate of the number of packets it will receive in this frame. Denote the actual number of packets by N and the estimate by $\hat{N}$. Analogous to CyMAC [14] which estimates data arrival interval by taking exponential weighted moving average of recent history data, CBA-EVT estimates N based on recent history data. However, it is known that exponentially weighted moving average does not do well when there is a trend in the data. Therefore, the online traffic estimation in CBA-EVT first detects whether or not recent history data has an increasing or decreasing trend. If a trend is detected, the estimate $\hat{N}$ is computed by Holt exponential smoothing [7]; otherwise, $\hat{N}$ is computed by exponentially weighted moving average.

The way to detect an increasing (decreasing) trend is to check whether or not the actual numbers of received packets in the past F frames are strictly increasing (decreasing), where F is a positive integer. If a trend is detected, $\hat{N}$ is computed by Holt exponential smoothing in the following way. Define $\{S_t\}$ as the smoothed value at time t, and $\{b_t\}$ as the best estimate of the trend at time t. The estimate $\hat{N}$ is given by the formulas:

$$\begin{aligned} &S_0 = N_0 \\ &b_0 = N_1 - N_0 \\ &S_t = k_\alpha N_t + (1-k_\alpha)(S_{t-1} + b_{t-1}) \\ &b_t = k_\beta(S_t - S_{t-1}) + (1-k_\beta)b_{t-1} \\ &\hat{N} = S_{F-1} + b_{F-1} \end{aligned} \tag{1}$$

where the index t is 0, 1, …, F-1, k_α is the data smoothing factor, $0 < k_\alpha < 1$, and k_β is the trend smoothing factor,

---

[1] Note that in this paper, the duration of a time slot is not short. For example, a time slot is of length 1 second in our simulation. Loose synchronization can be achieved, for example, by the techniques in [5], [6], [12], or [13].

$0 < k_\beta < 1$. In our simulation, k_α is set to 0.25 and k_β is set to 0.125.

2) Cost Benefit Analysis

The estimate $\hat{N}$ is fed into our cost benefit analysis which runs at the beginning of each frame to schedule reception slots. In CBA-EVT, reception slots (including hybrid slots) within a frame are set to be recurrent: s non-reception slots are followed by one reception slot; such *reception cycles* (*RCs*), each of length $s+1$ slots, repeat until the end of the frame, where the value of s is to be determined at run time. Take node X in Fig. 1 for example: the 1st frame contains three RCs with $s=1$ and the 2nd frame contains two RCs with $s=2$.

The value of s affects the two conflicting objectives, energy consumption and packet delay. This subsection describes the multi-objective optimization problem in the context of cost benefit analysis [10]. The next subsection presents how to obtain the optimal value s^* for each frame.

The key idea of applying the cost benefit analysis is to convert metrics of interest into *cost* and *benefit* and make decision based on the values of benefit and cost. Let us consider the benefit and the cost as functions of s. Denote the benefit function by $B(s)$ and the cost function by $C(s)$. The goal is to make nodes sleep as much as possible, as long as it is worth to sleep so much in the sense that the net benefit is not negative. Equivalently, the optimal value s^* is the largest integer such that $B(s^*) \geq C(s^*)$.

In the cost-benefit analysis, there must be a common unit of measurement. To this end, we set the units of cost and benefit functions both to be a unit of energy, Joule. We consider energy consumed on idle listening and state transition. Energy consumed on transmitting/receiving packets is not explicitly included in the benefit and cost functions because its value is independent of s provided that CSMA or other collision avoidance mechanisms make the occurrence of collision infrequent.

To transform metrics of interest other than energy into the same unit, we define two constants, β and ω. β converts the cost incurred by extra packet delay due to node's sleep into energy. ω converts the benefit earned by packet receptions into energy. The unit of β, ω, P_I, and P_{trans} are listed in Table 1.

TABLE 1
UNITS OF PARAMETERS IN THE CBA-BASED SCHEDULING

Parameter	Unit
β	Joule per packet per slot
ω	Joule per packet
idle listening energy, P_I	Joule per slot
state transition energy, P_{trans}	Joule per transition

The benefit and cost functions are defined in (2) and (3), respectively. The benefit function consists two terms. For explanation purpose, let us pretend that a node keeps awake all the time in its reception slots and keeps sleeping in its non-reception slots. The first term of (2) represents the reduction of energy consumed on idle listening. Because a node wakes up once every $s+1$ slots and a frame consists of L slots, the node sleeps $L - \lfloor L/(s+1) \rfloor$ slots within the frame and thus saves an energy of $P_I(L - \lfloor L/(s+1) \rfloor)$ compared to the case of keeping idle listening all the frame. The second term of (2) represents the benefit due to packet receptions. This benefit comes from one of the fundamental functions in a network—receiving and forwarding packets. In the second term, the estimate $\hat{N}$ is used instead of the actual number N because N is unknown at the beginning of the frame.

As shown in (3), the cost function consists of three terms. The first term represents the state transition energy from the sleep mode to the active mode, which is linear to the number of transitions $\lfloor L/(s+1) \rfloor$. The second and third terms involve the cost of extra packet delay incurred by node's sleep; the second term corresponds to the cost of extra packet delay for packets arriving in the $\lfloor L/(s+1) \rfloor$ complete reception cycles in the frame and the third term for packets arriving in the remaining $L-(s+1)\lfloor L/(s+1) \rfloor$ slots in the frame.

The second term of (3) is derived as follows. There are $\lfloor L/(s+1) \rfloor$ complete reception cycles in the frame. In each complete reception cycle, a node sleeps for s slots; thus the node will receive $s\frac{\hat{N}}{L}$ packets on average after it wakes up. Due to node's sleep, these packets have to wait for some time, ranging between 0 to s slots until the node wakes up; the average extra packet delay is $\frac{s}{2}$ slots. So node's sleep causes a cost of $\lfloor L/(s+1) \rfloor s\frac{\hat{N}}{L} \cdot \frac{s}{2} \cdot \beta = \beta \frac{\hat{N}}{2L} s^2 \lfloor L/(s+1) \rfloor$ in the complete reception cycles. Similarly, the third term can be derived.

So the multi-objective optimization problem in the context of cost benefit analysis can be formulated as follows:

Given:

Estimated number of packets in the frame, $\hat{N}$
Energy consumptions, P_I and P_{trans}
Frame length L
Constants, β and ω
The benefit function $B(s)$ and the cost function $C(s)$

$$B(s) = P_I\left(L - \left\lfloor \frac{L}{s+1} \right\rfloor\right) + \omega\hat{N} \tag{2}$$

$$C(s) = P_{trans}\left\lfloor \frac{L}{s+1} \right\rfloor + \beta\frac{\hat{N}}{2L}s^2\left\lfloor \frac{L}{s+1} \right\rfloor + \beta\frac{\hat{N}}{2L}\left(L - (s+1)\left\lfloor \frac{L}{s+1} \right\rfloor\right)^2 \tag{3}$$

Determine:

The largest integer $s^* \in \{0,1,\ldots,L-1\}$ such that $B(s^*) \geq C(s^*)$.

3) A Solver to the Cost Benefit Analysis and Speed-Up

To determine the optimal integer s^* in $\{0,1,\ldots,L-1\}$ in the cost benefit analysis mentioned above, a brute-force search algorithm is usually fast enough because of its complexity O(L). For further acceleration in the case when L is extremely large or when the computational power at sensor nodes is very low, the search algorithm can start at the point

$$s_0 = \left\lceil \frac{(P_I L + \omega\hat{N}) + \sqrt{(P_I L + \omega\hat{N})^2 + 2\beta\hat{N}(\omega\hat{N} - P_{trans}L)}}{\beta\hat{N}} \right\rceil$$

and search around the starting point for the largest integer at which the benefit is larger than the cost. This starting point s_0 is the positive root of $B(s) - C(s) = 0$, ignoring all the floor functions in $B(s)$ and $C(s)$. In the case when sensor nodes have extremely low computational power, a heuristic that choose $s^* = s_0$ can be used; this heuristic has complexity O(1).

B. *Dynamic Per-Slot Duty Cycle Adjustment*

The per-frame reception slot scheduling explained in Section III.A is designed, presuming that nodes are awake all the time in reception slots. An idea for further energy savings is to enter the sleep mode immediately after the completion time of each reception slot. Because it is impossible in event-based and contention-based WSNs to know the actual completion time beforehand, how to estimate the completion time is a practical challenge. To adapt to a wide range of dynamic network condition, we propose a mechanism of per-slot duty cycle adjustment consisting of two parts—EVT-based completion time estimation and hysteresis sleep. The former one specifies how to estimate the completion time; given the estimate, the latter one dictates when to sleep.

1) EVT-Based Completion Time Estimation

At beginning of a reception (and/or hybrid) slot, each node estimates the completion time of the slot. To save energy while not causing a significant extra delay, we utilize a conservative method: Instead of merely the next reception slot, we consider the next W reception slots (including hybrid slots) as a whole. Define $t_1, t_2, \ldots, t_W$ as the actual completion times of the coming W reception slots and assume that these W random variables follow an unknown distribution function. Our goal is to quantify the δ-reliable volume, $\hat{t}_\delta$, such that $\hat{t}_\delta$ is long enough for all of the coming W reception slots to receive all packets destined for the node with probability δ or larger. That is, $\Pr\{\max(t_1, t_2, \ldots, t_W) \le \hat{t}_\delta\} = \delta$ by definition. $\hat{t}_\delta$ is regarded as the estimate of the completion time of this reception slot.

We exploit extreme value theory [8] to get the value of $\hat{t}_\delta$, given the probability δ (*e.g.,* 90%). Denote the order statistics of $t_1, t_2, \ldots, t_W$ by $T_1, T_2, \ldots, T_W$. By extreme value theory, we get:

$$\hat{t}_\delta = T_{W-k} + \frac{T_{W-k} M_W^{(1)} (1-\hat{\gamma}_-)}{\hat{\gamma}} \left(\left(\frac{1}{1-\delta^{1/W}} \frac{k}{W} \right)^{\hat{\gamma}} - 1 \right) \tag{4}$$

where the intermediate number, k, is an integer parameter no greater than W. As in [9], the moment estimator $M_W^{(j)}$, the estimated extreme value index $\hat{\gamma}$, and $\hat{\gamma}_-$ are:

$$M_W^{(j)} = \frac{1}{k} \sum_{i=0}^{k-1} \left(\log T_{W-i} - \log T_{W-k} \right)^j, \; j = 1, 2$$

$$\hat{\gamma} = M_W^{(1)} + 1 - \frac{1}{2} \left(1 - \left(M_W^{(1)} \right)^2 / M_W^{(2)} \right)^{-1}$$

$$\hat{\gamma}_- = 1 - \frac{1}{2} \left(1 - \left(M_W^{(1)} \right)^2 / M_W^{(2)} \right)^{-1}$$

We give two remarks on this EVT-based method. First, this is a conservative method and it is possible to have an estimate $\hat{t}_\delta$ greater than the slot duration. In this case, $\hat{t}_\delta$ is truncated to be the slot duration. Second, the intermediate number k is a user-defined parameter, depending on memory size and computational power at a sensor node. Smaller memory size or lower computational power results in a smaller k.

2) Hysteresis Sleep

Given the estimated completion time $\hat{t}_\delta$ derived above, we propose a hysteresis sleep mechanism. As illustrated in Fig. 2, a node wakes up at beginning of a reception slot and keeps active for a time interval of length $\hat{t}_\delta$. If no packet destined for it arrives during this interval, the node enters the sleep mode immediately. Otherwise, the node extends the active state for a time interval of length Δ, which is called the hysteresis margin for sleeping decision. Backoff mechanism may defer sender's packet transmission; so CBA-EVT uses hysteresis margin to detect new packet without waiting for the next reception slot. If no packet destined for it arrives during this period of hysteresis margin, the node enters the sleep mode immediately. Otherwise, the sensor node extends the active state for another hysteresis margin. The active state can be extended many times, each for an interval of Δ, until no packet destined for the node arrives in the last hysteresis margin.

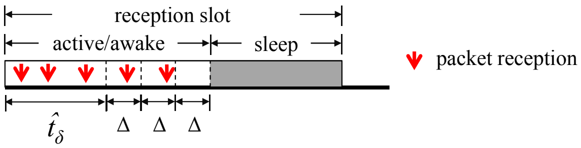

Fig. 2. Illustration of hysteresis sleep.

IV. EVALUATION

CBA-EVT is evaluated with the ns-2 simulator over a wide range of traffic load values. We consider the scenario where one node is the sink and other nodes are data sources. To focus on the impact of the MAC protocol in use, sensor nodes are deployed to form a cross. The node at the center is the sink. Any two adjacent nodes are 250 meters apart. There are 16 flows in total. CBA-EVT, PW-MAC, AMAC, and S-MAC are run respectively in the network to compare their performances.

To test adaptability to a wide range of online network condition, a wide range of traffic is injected into the network. Data packets generated at each source follows a Poisson arrival process. The average inter-arrival time of packet generation at each source varies from 0.1 second to 100 seconds. The end-to-end delay bound is set to ten seconds.

Common parameters in the simulation are showed in Table 2. Parameters are mostly set according to [11]. Remaining parameters are set according to the specification of TI's CC2430, an IEEE 802.15.4 compliant RF transceiver. A frame will be retransmitted up to a retry limit if no acknowledgement is received.

TABLE 2
SIMULATION PARAMETERS

Parameter	**Value**
Transmit power P_T	60 mW
Receive power P_R	12 mW
Idle power P_I	12 mW = 12 mJ per slot
Sleep power P_{sleep}	0.03 mW
State transition energy P_{trans}	0.03 mJ per transition
State transition time	5 ms
Packet length	50 bytes
Slot duration T	1 second
Hysteresis margin Δ	40 ms

Each tested MAC protocol has its own parameters as:

- S-MAC: Predetermined, fixed (per-slot) duty cycle is set to either 4%[2] or 10% to 100% with an increment of 5%.
- AMAC: Maximum cycle time is set to T, $2T$, $4T$, or $8T$, where T is the slot duration. The (per-slot) duty cycle is set to either 4% or 10% to 100% incremented by 5%.
- PW-MAC: The modulus m in the wake-up schedule generator is set to be 10 to 200 incremented by 10.
- CBA-EVT: L is set to 1, 2, 4, or 8. (That is, the frame duration is T, $2T$, $4T$, or $8T$.) β is 0.1, 10, or 100.

All combinations of parameters are evaluated. For each protocol, the most energy-efficient result among those that meet the delay bound is presented later in this section.

As shown in Fig. 3, CBA-EVT has lowest energy consumption, regardless the network traffic is low, medium, or high. This demonstrates that the CBA-EVT's run-time mechanisms, including CBA-based per-frame scheduling and EVT-based per-slot duty cycle adjustment, altogether adapt to a wide range of traffic load values effectively.

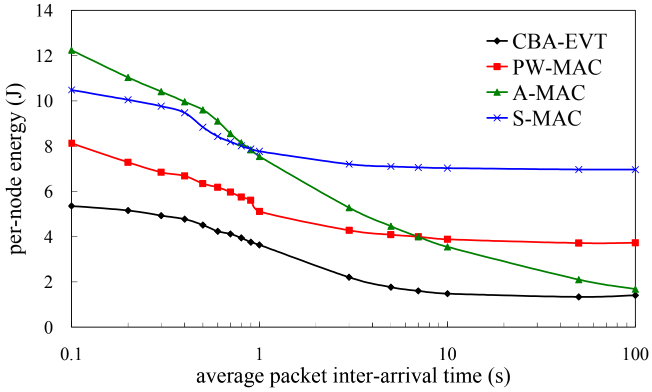

Fig. 3. Per-node energy consumption vs. average packet inter-arrival time.

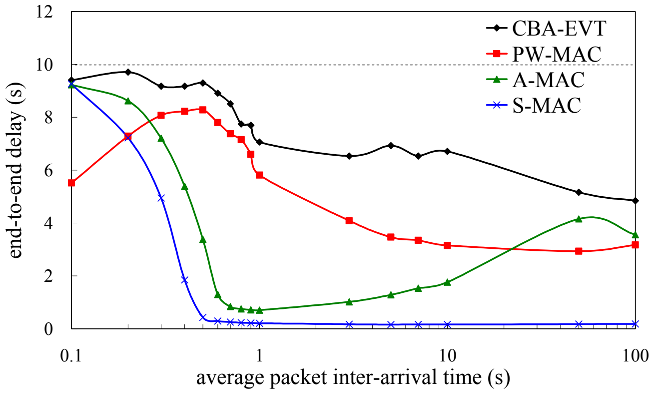

Fig. 4. End-to-end delay vs. average packet inter-arrival time.

More insight on adaptability of each tested protocol to online network condition can be inferred from Fig. 4. As observed, whereas the 1000 times increase in packet inter-arrival time decreases S-MAC's end-to-end delay by a factor of roughly 50 times, the same increase in packet inter-arrival time only halves CBA-EVT's end-to-end delay. S-MAC whose duty cycle remains fixed at run time wastes the opportunity of possible energy savings at a low traffic load, leaving a slack of almost 10 seconds in the delay budget. On the contrary, nodes in CBA-EVT (with a single set of system parameters) sleep more and save more energy, while the delay bound is still met.

To observe the impact of hop count to the sink on energy consumption for CBA-EVT, the energy consumption averaged over all nodes is decomposed into several components. As observed in Fig. 5, although each node has exactly the same system parameters, the per-node energy is obviously affected by the hop count. In the simulation setup, a node with a smaller hop count to the sink will consume more energy, because it helps more source nodes in forwarding their packets to the sink. From this point of view, Fig. 5 demonstrates the run-time adaptability of CBA-EVT (with a single set of system parameters) to traffic load variations.

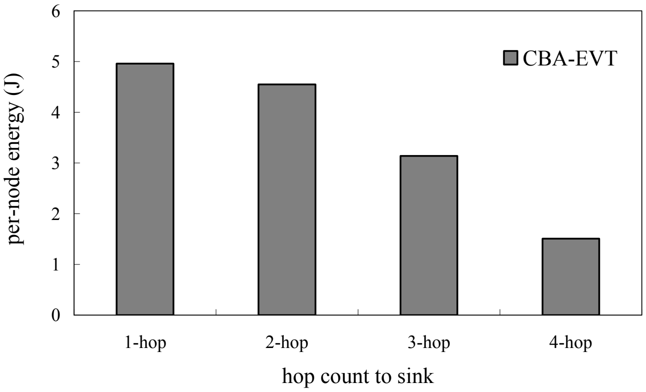

Fig. 5. Per-node energy consumption vs. hop count to the sink. The average inter-arrival time of packets generated at each node is one second.

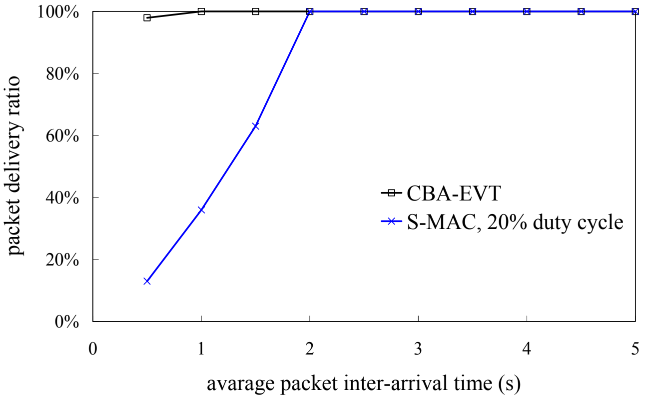

Fig. 6. Packet delivery ratio vs. average packet inter-arrival time.

Fig. 6 shows the packet delivery ratio of CBA-EVT, which is equal to (or almost equal to) 100%. This implies that CBA-EVT can accommodates the given traffic injection rates; meanwhile, CBA-EVT saves energy and meets the delay bound by dynamically adjusting the number of reception slots in each frame and the duty cycle in each reception slot. On the contrary, the S-MAC with 20% duty cycle protocol suffers a big variation in packet delivery ratio, despite it has a larger

---

[2] The reason why the minimum duty cycle is set to 4% is to match the hysteresis margin of 40 ms.

energy consumption than CBA-EVT at these given traffic injection rates.

V. CONCLUSION

In this paper, we have presented CBA-EVT, a novel energy-efficient MAC protocol that utilizes cost-benefit analysis and extreme-value theory. CBA-EVT is designed for a wide range of traffic load scenarios, whereas existing protocols work well only in a narrow range of operating points. CBA-EVT takes both energy conservation and packet delay into account. This fully distributed protocol estimates online traffic load and exploits the cost benefit analysis to schedule reception slots once a frame. To further save energy, CBA-EVT estimates the completion time of each reception slot by extreme-value theory and dynamically adjusts duty cycle on a per slot basis accordingly. Simulation results confirm that CBA-EVT is very effective in energy savings for a wide range of online network condition.

REFERENCES

[1] P. Lin, C. Qiao, and X. Wang, "Medium Access Control with a Dynamic Duty Cycle for Sensor Networks", in Proc. IEEE WCNC, 2004.

[2] W. Ye, J. S. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Network," in Proc. IEEE INFOCOM, 2002.

[3] W. Ye, J. S. Heidemann, and D. Estrin, "Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks," IEEE Transactions on Networking, vol. 12, no.3, pp. 493-506, March 2004.

[4] T. V. Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks", in Proc. ACM SenSys, 2003.

[5] S. H. Lee, J. H. Park, and L. Choi, "AMAC: Traffic Adaptive Sensor Network MAC Protocol through Variable Duty-Cycle Operations," in Proc. IEEE ICC, 2007.

[6] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "PW-MAC: An Energy-Efficient Predictive-Wakeup MAC Protocol for Wireless Sensor Networks," in Proc. IEEE INFOCOM, 2011.

[7] P. S. Kalekar, "Time series Forecasting using Holt-Winters Exponential Smoothing," Kanwal Rekhi School of Information Technology, Tech. Rep., 2004.

[8] L. D. Haan and A. Ferreira, *Extreme Value Theory: An Introduction*. Springer, 2006.

[9] A. L. M. Dekkers, J. H. J. Einmahl, and L. D. Haan, "A Moment Estimator for the Index of an Extreme-Value Distribution," Ann. Statist., vol. 17, no. 4, pp. 1833-1855, 1989.

[10] A. E. Boardman, D. H. Greenberg, A. R. Vining, and D. L. Weimer, *Cost-Benefit Analysis: Concepts and Practice.* Prentice Hall. 1996.

[11] Y.-S. Yen, S. Hong, R.-S. Chang, and H.-C. Chao, "An Energy Efficient and Coverage Guaranteed Wireless Sensor Network," in Proc. IEEE WCNC, March 2007.

[12] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, "Clock Synchronization for Wireless Sensor Networks: A Survey," Ad Hoc Networks, vol. 3, no. 3, May 2005.

[13] S. M. Lasassmeh and J. M. Conrad, "Time Synchronization in Wireless Sensor Networks: A Survey," in Proc. IEEE SoutheastCon, Charlotte-Concord, NC, March 2010.

[14] Y. Peng, Zi Li, D. Qiao, and W. Zhang, "Delay-Bounded MAC with Minimal Idle Listening for Sensor Networks," in Proc. IEEE INFOCOM, 2011.