Modeling and Throughput Analysis for X-MAC with a Finite Queue Capacity

Ou Yang, *Student Member, IEEE,* and Wendi B. Heinzelman, *Senior Member, IEEE*

***Abstract*—MAC layer duty-cycling is widely used to reduce idling listening, which is energy-intensive in wireless sensor networks. Among duty-cycling MAC protocols, asynchronous protocols generally have higher energy-efficiency and lower packet latency than synchronized ones. In this paper, we propose a Markov model to analyze the throughput of X-MAC, an asynchronous duty-cycled MAC protocol for wireless sensor networks. Simulations show that our analytical model provides throughput values that closely match the simulation results under various network conditions. Using our model to estimate the throughput of X-MAC is computationally efficient, providing accurate results quickly compared with simulations, which take a long time to run and require many iterations due to the large standard deviations in throughput. More importantly, our model enables designers to obtain a better understanding of the effect of different numbers of nodes and data arrival rates on the performance of X-MAC.**

***Index Terms*—X-MAC, Markov model, throughput analysis**

I. Introduction

Wireless sensor networks have many advantages including low cost and ease of deployment, and they support a wide range of applications, such as agricultural monitoring, emergency rescue, and military surveillance. However, energy-efficiency is one of the biggest impediments to the ubiquitous deployment of wireless sensor networks, since sensors are usually battery-powered and oftentimes it is impractical to change the batteries manually. Hence, saving energy to prolong the network lifetime is of paramount importance in wireless sensor networks.

One of the most widely used mechanisms for achieving low energy operation is duty-cycling at the MAC layer. Duty-cycling greatly reduces idle listening, which is energy intensive in wireless sensor networks [1]. There are two categories of duty-cycled MAC protocols for wireless sensor networks. One is synchronized protocols, like SMAC [1] and TMAC [2]. The other is asynchronous protocols, like B-MAC [3] and X-MAC [4]. Asynchronous duty-cycled MAC protocols remove the energy overhead for synchronization, and they are easier to implement as they do not require clock synchronization. However, while current IEEE 802.15.4 compliant platforms support asynchronous protocols like X-MAC, which uses a fixed preamble, asynchronous protocols like B-MAC, which uses variable-length preambles, can no longer be implemented on such platforms [5]. Hence, asynchronous protocols like X-MAC have more promising applications in wireless sensor networks. Many variations of fixed-preamble asynchronous duty-cycled MAC protocols have been proposed to improve energy efficiency and packet latency [6][7].

In this paper, we propose a Markov model to analyze the throughput of X-MAC under various network conditions. Throughput estimation is important for many applications, such as visual surveillance, that generate a large amount of data. Moreover, while there are many simulation studies for X-MAC [7][8], there are no existing analytical models for determining the throughput of X-MAC. We show that the throughput obtained from our analytical model matches simulation results within 5%. Using our analytical model to obtain the throughput of X-MAC not only removes the need for large sets of simulations, which are time consuming, but it also provides more accurate results, since the simulated throughput has large standard deviation.

The rest of this paper is organized as follows. Section II introduces the X-MAC protocol. Section III presents the proposed Markov model, and Section IV elaborates on the throughput analysis of X-MAC using the Markov model. Section V validates our analytical model. Section VI discusses related work, and Section VI concludes the paper.

II. Overview of X-MAC

X-MAC is an asynchronous duty-cycled MAC protocol for wireless sensor networks. X-MAC avoids synchronization overhead, and hence it has higher energy-efficiency than synchronized MAC protocols. Additionally, X-MAC uses a series of short preamble packets instead of an extended preamble, like B-MAC. The short preamble packets carry the address information of the destination node. As a result, non-destination nodes can go to sleep as soon as they hear the first short preamble instead of remaining awake until the extended preamble ends. Moreover, the destination node can reply with an ACK in between two successive short preambles to stop the preamble and start the data transfer. Therefore, this strobed preamble approach saves energy and greatly reduces latency. Furthermore, X-MAC has a fixed preamble size, and hence it can be readily adapted to the packetized radios that are emerging as the standard in today's sensor motes [4].

Fig. 1 shows the timeline of X-MAC. Every node in the network wakes up periodically to send and receive packets. The interval between two successive wake-ups is a cycle. X-

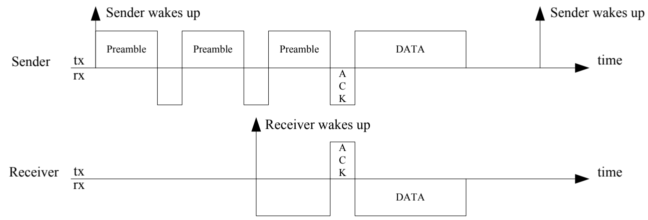

Fig. 1. X-MAC timeline.

MAC has a fixed cycle length T for every node, yet each node starts its duty cycle with an arbitrary offset. As a result, when a sender wakes up to send a packet, the receiver may still be sleeping. Hence, the sender, from the time it wakes up, starts sending short preamble packets with the receiver's address information. In between two successive preamble packets, the sender pauses to listen to the media. Non-receiver nodes hear the preamble and go to sleep until the next wake-up time. At some point, the receiver wakes up and hears the preamble. The receiver sends an acknowledgement back to the sender during the pause between the two preambles. Note that the pause is shorter than the time that a node needs to detect an ongoing transmission, hence only the receiver can access the channel during the pause, while other nodes cannot interfere with the communication between the sender and the receiver [5]. When the sender receives the acknowledgement, it starts sending the DATA packet as the receiver is ready to receive.

Since X-MAC is asynchronous, for each DATA packet that is successfully delivered, the average communication time for the sender is $T/2$ plus the length of a DATA packet L_{DATA} [5]. X-MAC also has collisions. When more than one sender wake up and start sending their preambles at the same time, all the other nodes, including the receivers, cannot determine the destination address information in the preambles. In this case, the senders will not stop sending preambles until their next wake-up time. Hence, for each colliding DATA packet, the average communication time for the sender is T [5]. For simplicity, in this paper we analyze the throughput of slotted X-MAC. Hence T and L_{DATA} are in the units of time slots, τ.

III. Markov Model of X-MAC

As has been done for previous MAC protocols such as IEEE 802.11 [10] and SMAC [9][16], we propose a Markov model to describe the behavior of X-MAC. Using this model, the throughput of X-MAC can be obtained under various network conditions. The proposed Markov model assumes (1) independent packet arrivals at each node, (2) finite queue capacity at each node, (3) ideal channel (no fading, no capture effect, and no hidden terminals), (4) one transmission opportunity or one DATA packet reception per cycle at each node, (5) no retransmissions, and (6) every node has a constant probability of transmitting a DATA packet regardless of any node's queue length (similar assumptions were made in [10][11], and were verified as good approximations of real scenarios).

Fig. 2 shows the proposed Markov model for X-MAC.

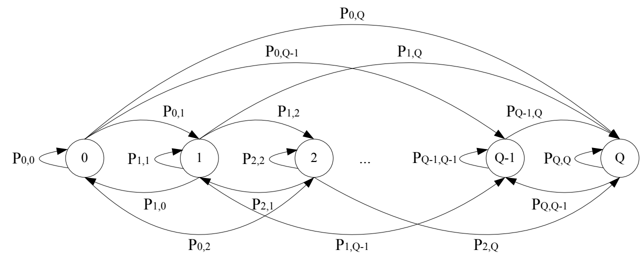

Fig. 2. Markov model for X-MAC.

Assume the queue capacity is Q at each node. The Markov model has Q+1 states, from left to right corresponding to 0 packets in the queue to Q packets in the queue (full queue). A node may change its status cycle by cycle, corresponding to the transition from one state to another in the Markov model. The transition probabilities from one state to another can be represented as follows:

$$P_{0,i} = A_i, i = 0..Q-1 \tag{1}$$

$$P_{0,Q} = A_{\geq Q} \tag{2}$$

$$P_{i,i-1} = p \cdot A_0, i = 1..Q \tag{3}$$

$$P_{i,j} = p \cdot A_{j-i+1} + (1-p) \cdot A_{j-i}, i = 1..Q-1, j = i..Q-1 \tag{4}$$

$$P_{i,Q} = p \cdot A_{\geq Q-i+1} + (1-p) \cdot A_{\geq Q-i}, i = 1..Q \tag{5}$$

$$P_{i,j} = 0, i = 2..Q, j = 0..i-2 \tag{6}$$

where, A_i is the probability that i DATA packets arrive at a node in a cycle, $A_{\geq i}$ is the probability that no less than i DATA packets arrive at a node in a cycle, and p is the probability for a node to transmit a DATA packet in a cycle.

Specifically, equations (1) and (2) show that transitions from the empty queue state to any other states depend only on the new packet arrivals. Equations (3) and (6) show that only one DATA packet can be transmitted during a cycle with probability p. Equations (4) and (5) show that a transition with a non-decreasing queue has two possible cases, depending on whether the oldest DATA packet in the queue wins the media (first term) or not (second term). Equations (2) and (5) show that packets are dropped when the queue overflows.

IV. Throughput Analysis of X-MAC

Throughput is defined as the amount of data successfully delivered within a unit time. For X-MAC, the throughput can be calculated within a cycle time, since nodes work in a duty-cycled fashion. Assuming there are N nodes in the network, a DATA packet has size S, a cycle has T slots, the length of a slot is τ, and the probability for each node to successfully deliver a DATA packet in a cycle is p_s, if the proposed Markov model has a unique stationary distribution $\pi = (\pi_0, ..., \pi_Q)$, the throughput of X-MAC is

$$THR = N \cdot (1-\pi_0) \cdot p_s \cdot S/(T \cdot \tau). \tag{7}$$

Since N, S, T, and τ are known, only π_0 and p_s need to be obtained to calculate throughput. π_0 and p_s are determined by (1) X-MAC's duty-cycle behavior, which is described by our proposed Markov model, and (2) the media access rules of X-MAC. Hence, we first examine the proposed Markov model, and then we analyze the media access rules of X-MAC.

A. Examine the Markov Model of X-MAC

The proposed Markov model has a state space $\widetilde{S}=\{0,1,...,Q\}$ and a transition matrix P. Since the Markov model is irreducible and aperiodic, it has a unique stationary distribution $\pi=(\pi_0,...,\pi_Q)$ such that

$$\pi_i \geq 0 \text{ for any } s_i \in \widetilde{S}, \quad \sum_{s_i\in\widetilde{S}} \pi_i = 1, \quad \pi P = \pi \tag{8}$$

Assuming a-priori data arrival knowledge (A_i and $A_{\geq i}$ are known), the probability for each node to transmit a DATA packet p is the only unknown variable in the transition matrix P. Thus, the stationary distribution of the Markov chain $\pi=(\pi_0,...,\pi_Q)$ can be obtained as a function of p. Specifically, we define a function $f(\cdot)$, such that

$$\pi_0 = f(p). \tag{9}$$

Hence, for a given probability for each node to transmit a DATA packet p, the stationary probability of the empty queue state π_0 can be obtained from the Markov model. If we can find another relationship between p and π_0 from the media access rules of X-MAC, we can use these relationships between p and π_0 to obtain the throughput of X-MAC.

B. Examine the Media Access Rules of X-MAC

Define p_f to be the probability of a collision when a node transmits a DATA packet in a cycle. Hence

$$p = p_s + p_f. \tag{10}$$

Define A to be the event that a node successfully delivers a DATA packet in a cycle, B to be the event that a node has collisions when transmitting a DATA packet in a cycle, *free* and *busy* to be the status of the channel, and *empty* and $\overline{empty}$ to be the status of the queue at a node. Therefore, according to the Markov model, we have

$$\begin{aligned} p_s &= \Pr(A, free \mid \overline{empty}) \\ &= \Pr(A \mid free, \overline{empty}) \cdot \Pr(free \mid \overline{empty}) \end{aligned}, \tag{11}$$

$$\begin{aligned} p_f &= \Pr(B, free \mid \overline{empty}) \\ &= \Pr(B \mid free, \overline{empty}) \cdot \Pr(free \mid \overline{empty}) \end{aligned}. \tag{12}$$

We first solve for $\Pr(A \mid free, \overline{empty})$ and $\Pr(B \mid free, \overline{empty})$, and then we determine $\Pr(free \mid \overline{empty})$.

Given that a node has packets in its queue, and that the channel is free when it wakes up, the node can successfully transmit a DATA packet as long as (1) no other nodes in the network wake up at the same time, or (2) some nodes wake up at the same time, but they have no packets to send. Hence,

$$\Pr(A \mid free, \overline{empty}) = \sum_{t=1}^{T} \frac{1}{T} \left(\sum_{i=0}^{N-1} \binom{N-1}{i} \left(\frac{1}{T}\right)^i \pi_0^{\,i} \left(\frac{T-1}{T}\right)^{N-1-i} \right). \tag{13}$$

Similarly, given a node has packets to send in the queue, and the channel is free when it wakes up, the node has a collision when transmitting a DATA packet if at least one other node in the network wakes up at the same time and has packets to send. Hence,

$$\Pr(B \mid free, \overline{empty}) = \sum_{t=1}^{T} \frac{1}{T} \left(\sum_{i=1}^{N-1} \binom{N-1}{i} \left(\frac{1}{T}\right)^i \left(\sum_{j=1}^{i} (1-\pi_0)^j \pi_0^{\,i-j} \right) \left(\frac{T-1}{T}\right)^{N-1-i} \right). \tag{14}$$

$\Pr(free \mid \overline{empty})$ is the probability of a free channel when a node wakes up with packets to send in its queue. Since in X-MAC every node wakes up and sends packets with an arbitrary offset to other nodes, the channel experiences the same probability to be *free* or *busy* in every time slot. Hence, when a single node wakes up, no matter whether its queue is empty or not, the node sees the channel with the same probability of being *free* or *busy*. Therefore,

$$\Pr(free) \approx \Pr(free \mid \overline{empty}). \tag{15}$$

This approximation is validated by comprehensive simulations using Matlab. Consequently, the problem of determining $\Pr(free \mid \overline{empty})$ thus becomes the problem of determining $\Pr(free)$.

The probability of a free channel, $\Pr(free)$, can be obtained if the following two parameters are known. (1) the average length of a free channel between two transmissions over the media, E_{free}, and (2) the average length of a busy channel between the two chunks of a free channel, E_{busy}.

$$\Pr(free) = \frac{E_{free}}{E_{free} + E_{busy}}. \tag{16}$$

To calculate E_{free}, consider the time instant when a transmission ends and a chunk of free channel begins (note that the length of a specific chunk of free channel could be zero). From that time instant, the channel could be free for a certain number of cycles, say n cycles, until in the $n+1$ st cycle, some node(s) start to transmit at the t th slot. The length of this chunk of free channel is $n \cdot T + t$, and the probability of this event $P_{free}(n,t)$ can be obtained as

$$P_{free}(n,t) = \sum_{i=0}^{N-1} \sum_{j=1}^{N-i} \sum_{k=1}^{j} \binom{N}{i} \left(\frac{t}{T}\right)^i \pi_0^{\,i} \binom{N-i}{j} \left(\frac{1}{T}\right)^j \binom{j}{k} (1-\pi_0)^k \pi_0^{\,j-k} \left(\frac{T-t-1}{T}\right)^{N-i-j} \tag{17}$$

Therefore, the average length of a free channel between two transmissions over the media

$$E_{free} = \sum_{n=0}^{\infty} \sum_{t=1}^{T} (nT+t) \cdot P_{free}(n,t). \tag{18}$$

Similarly, when the channel is free for n cycles and t slots, the probability that a transmission is successful is $P_{busy}^{suc}(n,t)$, and the probability that a collision occurs is $P_{busy}^{col}(n,t)$.

$$P_{busy}^{suc}(n,t)=\sum_{i=0}^{N-1}\sum_{j=1}^{N-i} \binom{N}{i}\left(\frac{t}{T}\right)^{i}\pi_0^{\,i}\binom{N-i}{j}\left(\frac{1}{T}\right)^{j}\binom{j}{1}(1-\pi_0)\pi_0^{\,j-1}\left(\frac{T-t-1}{T}\right)^{N-i-j} \quad (19)$$

$$P_{busy}^{col}(n,t)=\sum_{i=0}^{N-2}\sum_{j=2}^{N-i}\sum_{k=2}^{j} \binom{N}{i}\left(\frac{t}{T}\right)^{i}\pi_0^{\,i}\binom{N-i}{j}\left(\frac{1}{T}\right)^{j}\binom{j}{k}(1-\pi_0)^{k}\pi_0^{\,j-k}\left(\frac{T-t-1}{T}\right)^{N-i-j} \quad (20)$$

The average length of a busy channel between the two chunks of a free channel E_{busy} can be calculated as the average length of a successful transmission $T/2+L_{DATA}$ times the probability of a successful transmission, plus the length of a colliding transmission T times the probability of a collision. Hence,

$$E_{busy}=\sum_{n=0}^{\infty}\sum_{t=1}^{T}\left((T/2+L_{DATA})\cdot P_{busy}^{suc}(n,t)+T\cdot P_{busy}^{col}(n,t)\right). \quad (21)$$

Plugging equations (18) and (21) into (16), we obtain $\Pr(free)$. Then plugging equation (16) into (15), we obtain $\Pr(free\,|\,\overline{empty})$. Plugging equation (13) and (15) into (11), and (14) and (15) into (12), the probability for each node to successfully transmit a DATA packet p_s, and the probability for each node to encounter a collision p_f are solved as a function of the stationary probability of the empty queue state π_0. According to equation (10), the probability for each node to transmit a DATA packet p can also be obtained as a function of π_0. Let function $h(\cdot)$ describe the relationship between p and π_0, we have

$$p=h(\pi_0)\,. \quad (22)$$

Solving the two equations (9) and (22), the actual p and π_0 under which X-MAC is operating for a given scenario can be determined. In Fig. 3 the blue curve shows equation (9) obtained from the Markov model, while the red dashed curve shows equation (22) obtained from the media access rules of X-MAC. The solution of the two functions $(p^*,\pi_0^{\,*})$ is also shown as the intersection of the two curves in the figure. Plugging in the actual $\pi_0^{\,*}$ to equation (11), the probability for each node to successfully deliver a DATA packet p_s can be obtained. The throughput of X-MAC can therefore be solved according to equation (7).

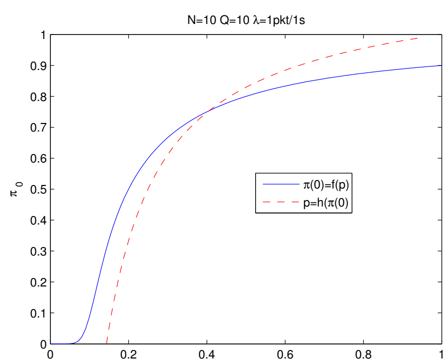

Fig. 3 Functions $f(\cdot)$ and $h(\cdot)$ used to solve for π_0 and p.

V. Simulation Results

In this section, we compare the analytical throughput obtained from equation (7) with the throughput obtained from Matlab simulations under varying numbers of nodes and varying data arrival rates. We assume: (1) nodes are fully connected and the channel is ideal, (2) each node randomly selects a destination among its neighbors for each packet to send (routing is not considered to exclude influences from other than X-MAC), (3) data arrive at each node according to a Poisson process, (4) queue capacity Q at each node is 10, (5) MAC layer DATA packet size S is 50 bytes, (6) the length of a cycle T is 100 slots and the slot size is 1ms, and (7) the time used to transmit a DATA packet L_{DATA} is 5 slots. Our simulation results are averaged over 1000 runs. Each run simulates X-MAC for 90s.

A. Varying the Number of Nodes

In this experiment, we vary the number of nodes in the network from 2 to 30. The data arrival rate at each node is 1pkt/s. Fig. 4 shows the throughput obtained from simulations and from our throughput analysis using the Markov model. Our analytical results match the simulation results with less than 5% difference in throughput. In Fig. 4, the throughput increases linearly as the number of nodes increases. Within this linear increasing section, X-MAC can deliver all the arriving DATA packets, hence the larger the number of nodes in the network, the higher the throughput X-MAC can achieve. However, as the number of nodes increases to a certain point, the throughput starts to decrease. Since a sender takes on average $T/2+L_{DATA}$ to finish sending a DATA packet, X-MAC has a delivery limit. When the throughput gets closer to this limit, as the number of nodes increases, more collisions occur, and as a result the throughput decreases. Fig. 4 also shows that as the number of nodes increases, the simulation

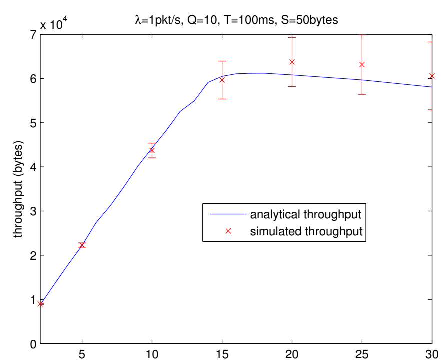

Fig. 4 Throughput of X-MAC with varying number of nodes (1 pkt/s).

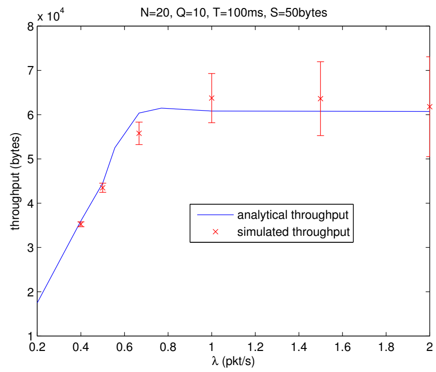

Fig. 5 Throughput of X-MAC with varying data arrival rate (20 nodes).

results have larger standard deviations. Hence, a few runs of the simulation become less representative in evaluating the performance of X-MAC. Instead, our analysis can accurately predict the throughput of X-MAC even when the number of nodes is large.

B. *Varying the Data Arrival Rate*

In this experiment, we vary the data arrival rate at each node from 0.2pkt/s to 2pkt/s. There are 20 nodes in the network. Fig. 5 shows that our analytical throughput matches the throughput obtained from the simulations. When the data arrival rate is small, the throughput of X-MAC increases linearly as the data arrival rate increases, since X-MAC can deliver all the arriving traffic. However, the throughput remains the same after X-MAC reaches its delivery limit. From this point, the queue at each node overflows, and every node has a packet to send whenever it wakes up. Therefore, when the date arrival rate further increases, the contention situation in the network does not change as extra arriving packets are dropped by the queue. Hence, the throughput of X-MAC remains the same. Again, Fig. 5 shows that as the data arrival rate increases, the standard deviations of the simulations increase. Hence, using simulations to evaluate the performance of X-MAC is not only time-consuming, but requires large numbers of iterations to produce meaningful results. Our throughput analysis using the Markov model can accurately solve this problem.

VI. Related Work

Although no existing work evaluates the throughput of X-MAC analytically, Rousselot et al. proposed a radio model to calculate the lower bound of X-MAC's power consumption [12]. Meanwhile, there has been previous work utilizing a Markov model to describe the behavior of other MAC protocols. Specifically, we modeled and analyzed the throughput of SMAC, a synchronized duty-cycled MAC protocol for wireless sensor networks [9]. A similar Markov model was used to describe the behavior of SMAC. However, since SMAC is synchronized and it has different rules for media access compared to X-MAC, the throughput analysis is unique for SMAC and X-MAC. Y. Zhang also proposed a Markov model to analyze the performance of SMAC [15], however, the design of the Markov model was fundamentally different with the one proposed in [9]. In an area other than wireless sensor networks, Bianchi proposed a Markov model to analyze the saturation throughput of IEEE 802.11 with unlimited retransmissions [13]. Fallah et al. [11] and He et al. [14] also proposed Markov models to analyze the throughput of IEEE 802.16 and IEEE 802.15.4 networks, respectively.

VII. Conclusions and Future Work

In this paper we modeled and analyzed the throughput of X-MAC with a finite queue capacity. The analytical results are validated by simulation results. The proposed Markov model and throughput analysis can be used to estimate the throughput of X-MAC under various node densities and traffic loads. Our future work includes using the proposed model to analyze the latency and energy consumption of X-MAC, and extending the model to multi-hop networks.

Acknowledgment

This work was supported in part by NSF #CNS-0448046.

References

[1] Y. Wei, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in Proc. INFOCOM '02, 2002, pp. 1567-1576.
[2] T. Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks", in Proceedings of the 1st Int. Conf. on Embedded Networked Sensor Systems, 2003, pp. 171-180.
[3] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in Proc. SenSys'04, 2004, pp. 95–107.
[4] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in Proc. SenSys'06, 2006, pp. 307–320.
[5] C. Merlin, "Adaptability in Wireless Sensor Networks Through Cross-Layer Protocols and Architectures", Ph.D. Dissertation, Dept. ECE, University of Rochester, Rochester, NY, 2009.
[6] K.-J. Wong and D. Arvind, "SpeckMAC: Low-power decentralised MAC protocol low data rate transmissions in Specknets," in Proc. REAL-MAN'06, 2006.
[7] S. Hong and H. Kim, "A Multi-hop Reservation Method for End-to-End Latency Performance Improvement in Asynchronous MAC-based Wireless Sensor Networks", IEEE Trans. Consumer Electronics, vol. 55, no. 3, Aug. 2009, pp. 1214-1220.
[8] H. Wang, X. Zhang, F. Naït-Abdesselam, and A. Khokhar, "DPS-MAC: An Asynchronous MAC Protocol for Wireless Sensor Networks", in Proc. HiPC'07, 2007, pp. 393-404.
[9] O. Yang and W. Heinzelman, "Modeling and Throughput Analysis for SMAC with a Finite Queue Capacity", in Proc. ISSNIP 09', 2009, pp. 409-414.
[10] K. Ghaboosi, B. H. Khalaj, Y. Xiao, and M. Latva-aho, "Modeling IEEE 802.11 DCF using parallel space-time Markov chain", IEEE Trans. Vehicular Technology, vol. 57, no. 4, Jul. 2008, pp. 2404-2413.
[11] Y. Fallah, F. Agharebparast, M. Minhas, H. Alnuweiri, and V. Leung, "Analytical modeling of contention-based bandwidth request mechanism in IEEE 802.16 wireless networks", IEEE Trans. Vehicular Technology, vol. 57, no.5, Sep. 2008, pp. 3094-3106.
[12] J. Rousselot, A. El-Hoiydi, and J. Decotignie, "Low Power Medium Access Control Protocols for Wireless Sensor Networks", in Proc. of the 14th European Wireless Conference, 2008, pp.1-5.
[13] G. Bianchi, "IEEE 802.11 – saturation throughput analysis", IEEE Communication Letters, vol. 1, no. 12, Dec. 1998, pp. 318-320.
[14] J. He, Z. Tang, H. Chen, and Q. Zhang, "An accurate and scalable analytical model for IEEE 802.15.4 slotted CSMA/CA networks", IEEE Trans. Wireless Communications, vol. 8, no. 1, Jan. 2009, pp. 440-448.
[15] Y. Zhang, C. He, and L. Jiang, "Energy and QoS Trade-off Analysis of S-MAC protocol in Wireless Sensor Networks", in Proc. CCWMSN'07, 2007, pp. 76-79.