Energy Efficient Multi-Layer Based Flat MAC Protocol for Wireless Sensor Networks

Ambresh G. Biradar
Department of Electronics and Communication Engineering
Jyothy Institute of Technology
Bangalore, India

N. S. V. Shet
Department of Electronics and Communication Engineering
National Institute of Technology, Karnataka
Surathkal, India

Abstract— **Energy efficiency is a major consideration while designing wireless sensor network nodes. The proposed Multilayer MAC (ML-MAC) protocol uses periodic listen and sleep modes to save energy. By a simulation with Matlab, we evaluated our protocol. ML-MAC is attempted to reduce the energy consumption beyond the S-MAC and T-MAC by reducing the idle listening and number of collisions.**

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

I. INTRODUCTION

In sensor network, sensor nodes are deployed in an ad-hoc manner to communicate using short-range radio channel [1]. Energy efficiency is the kernel issue in the designing of wireless sensor network MAC protocols. Most sensor network applications require energy autonomy for the complete lifetime of the node, which may span up to several years. These energy constraints require that the system be built such that Wireless sensor networks use battery-operated computing and sensing devices [2]. Therefore, designing energy-efficient communication protocols is important in wireless sensor networks. A MAC protocol is required in sensor networks to coordinate the sensor nodes' access to the shared medium. Designing power efficient MAC protocols is one of the ways to prolong the lifetime of the network. Commercial standards like IEEE 802.11 have a power management scheme for ad-hoc networks, wherein the nodes remain in idle listening state at low traffic to conserve power. Studies show that significant power is wasted even in the idle listening mode. Hence, 802.11 is not suitable for sensor networks [3].

Sensor nodes in proposed Multilayer MAC protocol are allocated into different layers so that the listen periods of the different layers are non-overlapping. Distributing the nodes among the layers allows the nodes to sleep longer and conserve battery power. This distribution also effectively reduces the traffic seen by the nodes in each layer, which reduces the probability of collisions and retransmissions.

II. RELATED WORK

Some applications are real time where latency is an important design factor. However, most of the applications of wireless sensor networks to date are not as sensitive to delay as they are to power consumption [4]. The following factors are important because they serve as a guideline to design a protocol or an algorithm for sensor networks [1]: Power consumption, Scalability, Fault Tolerance, Sensor Network Topology, Environment, Hardware Constraints, and Type of communication. Sensor networks have to be power efficient and scalable, whereas throughput, latency and fairness are the main points in normal wireless networks that are designed for voice or data in order to provide high Quality of Service (QoS) [5]. A tradeoff can be made between power consumption and others constraints that are not important for wireless sensor networks such as throughput, delay and fairness.

Current MAC design for wireless sensor networks can be broadly classified into two categories: contention-based protocols and reservation-based (TDMA) protocols [6]. Each of these access methods has its own advantages and disadvantages. In reservation-based MAC protocols, the channel is reserved for the nodes for a certain amount of time. Reservation based protocol have no collisions, it requires an efficient time schedule, clock synchronization and adjusting to network topology changes. However, TDMA protocols are not as scalable as contention-based protocols [6, 7].

On other hand, nodes in contention-based MAC protocols determine if they can access the medium by sensing the shared channel and competing to get access to it instead of defining schedules for access. Contention-based protocol is promising in terms of its simplicity, flexibility and robustness, it will suffer from contention, idle listening, overhearing, and this will waste lots of energy [7]. Therefore, many researchers are trying to propose new contention-based MAC protocols that overcome these sources of energy inefficiency. The IEEE 802.11 [8] is an international standard of physical and MAC layer specifications for wireless networks. It uses CSMA/CA. It is a simple and reliable MAC protocol that is widely used in many traditional ad-hoc wireless networks. However, it is not suitable for sensor networks because throughput, latency, and fairness were the primary design criteria, not power consumption. However, because of its simplicity and reliability, many researchers are trying to modify and develop the IEEE 802.11 so that it is applicable for wireless sensor networks. S-MAC [9, 10] is a contention-based approach that modified the IEEE 802.11 standard to be suitable for sensor networks. Each node sleeps for some time, and then wakes up and listens to see if any other node wants to talk to it. During sleeping, the node turns off its radio, and sets a timer to awake it later. Communication occurs only in the active (listen) period. Packets that are generated during the sleep period are buffered for the next frame cycle. This increases the latency because the sender has to wait for the active period [2].

S-MAC tries to reduce energy consumption from all of the sources of energy consumption. In S-MAC, nodes try to form one cluster by following the same listen/sleep schedule, i.e., by listening and sleeping at the same time, so that broadcasts need only be transmitted once and also the number of control packets are decreased. Therefore, S-MAC eliminates clustering to reduce inter-cluster communication and interference, but at the expense of making the listening period longer. When a new node in S-MAC joins the network, it first waits for a certain amount of time to get a schedule from another node and then follows that schedule. If it does not hear any schedule, it chooses its own and broadcasts it to the other nodes. After that, all nodes broadcast their schedules periodically using the control packets SYNC which have the time of the next frame cycle. Using this scheme a node can easily join the network.

III. PROPOSED ML-MAC PROTOCOL

A. Design overview of ML-MAC

A Multi-Layer MAC protocol is proposed as a technique to reduce node power consumption beyond that achieved by S-MAC [2, 9] and T-MAC [10]. ML-MAC is a distributed contention- based MAC protocol where nodes discover their neighbors based on their radio signal level. ML-MAC is a self-organizing MAC protocol that does not require a central node to control the operation of the nodes.

As Fig. 1 shows, time in ML-MAC is divided into frames and each frame is divided into two periods: listen and sleep. The listen period is also divided into 4 non-overlapping layers. Nodes are distributed among this set of layers where nodes in each layer follow a listen and sleep schedule that is skewed in time compared to the schedules of the other layers. Therefore, the listen periods of the nodes in different layers are non-overlapping. A node in ML-MAC protocol wakes up only at its assigned layer. Therefore, ML-MAC requires a lesser amount of energy than S-MAC because the listen period of a node in ML-MAC is shorter than the listen period of the frame in S-MAC. Main advantages of adopting multiple layers in ML-MAC are: Reduced energy consumption, Low average traffic and Extended network lifetime. Making the listening period shorter increases the probability that a node generates a packet while it is in the sleep mode. Those packets are buffered for transmission during an upcoming listen period. This results in a longer packet delay. However, since delay is not a primary design factor, this effect is acceptable in most sensor network applications [11].

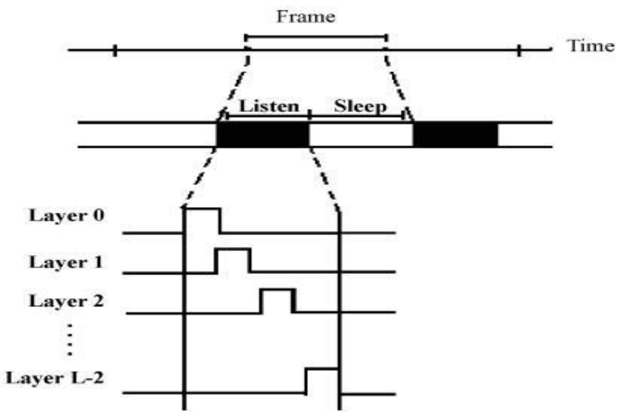

Fig. 1. Design overview of ML-MAC

In ML-MAC, any packet needing to be sent is initially assigned a reservation slot between 0 and 7 as in the IEEE 802.11 Distributed Control Function (DCF) implementation. If there is a collision, then the collided packets are assigned another reservation slot but this time the number of reservation slots is doubled from 8 to 16. To minimize the number of collisions, the number of reservation slots is doubled each time a collision occurs up to the 8th retransmission which would increase the number of reservation slots to 256. This is called binary exponential back-off.

Upon deployment, all the nodes are allocated schedules randomly. When a new node joins the network later, it chooses a schedule randomly and broadcasts its schedule to other nodes and all the nodes are allocated schedules randomly. In order for any node in the network to be aware of the listening time of other nodes in different layers, each node maintains a schedule table to store the schedules of all other nodes. As in S-MAC [2], nodes broadcast their schedules to other nodes using a control packet called SYNC. Also, to prevent clock drift, nodes update their schedules by periodically exchanging the control packet SYNC. This control packet is very short and it has the node number and the number of its access layer with the time of its next listen period.

B. Design considerations

A network application has Network lifetime T_N is divided into N_f frames and has the following design specifications.

n: Total number of nodes in the network.

λ: Average packet rate per node.

T_N: Network lifetime.

T_R: Maximum response time delay.

τ_t: Packet transmission delay.

τ_p: Propagation delay.

τ_d: Clock drift delay.

C: Battery capacity.

ρ: Average node power consumption.

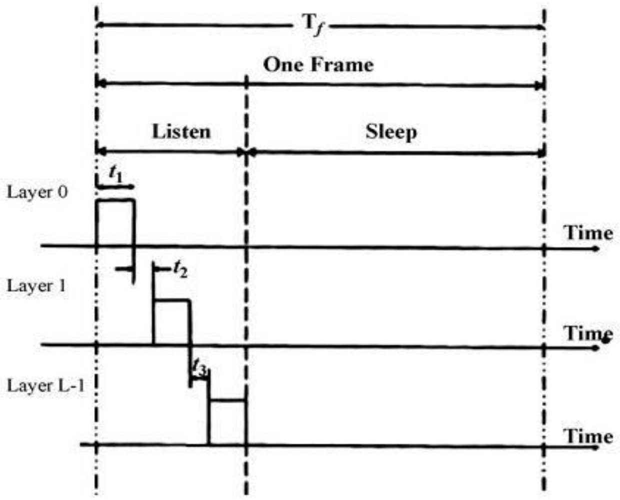

Fig. 2. Timing parameters of ML-MAC: layer duration tl, dead time between successive layers t2, and frame duration t3.

In addition, as shown in Fig. 2, the design parameters of ML-MAC include the following:

L: Number of access layers.

T_f : Frame duration.

N_f : Number of frames.

t_1 : Layer duration.

t_2 : Dead time between layers.

C. Design Procedure

Step1(Calculating the frame duration): For a given maximum response time delay T_R that is governed by the time to respond to and report events, the frame duration is bounded from above by:

$$T_f \leq T_R \tag{1}$$

T_f is also bounded from below by total listening time t_1 for all the layers:

$$T_f > t_1 \times L \tag{2}$$

Where t_1 is the listening period for one layer that will be found from step 2 and L is the number of layers.

Thus, the number of frames N_f is also bounded by:

$$\frac{T_N}{T_R} \leq N_f < \frac{T_N}{t_1 \times L} \tag{3}$$

Step 2 (Calculating the layer duration t_1) the duration of the listening period for one layer t_1 is governed by the battery capacity C and the average node power consumption ρ:

$$\rho \times t_1 \times N_f \ \leq \ C \times \vartheta \tag{4}$$

Where ϑ is the average output voltage of the battery.
Thus, t_1 is bounded from above by:

$$t_1 \leq \frac{C \times \vartheta}{\rho \times N_f} \tag{5}$$

Also, t_1 is bounded from below by the time needed to send at least one packet which is given by the following equation:

$$t_1 > \tau_t + \tau_p + 2\tau_d + W\tau_p \tag{6}$$

Where τ_t is the packet transmission delay, τ_p is the propagation delay, τ_d is the clock drift delay, and W is the maximum number of reservation slots which is called the window size. Thus, t_1 is bounded by:

$$\tau_t + \tau_p + 2\tau_d + W\tau_p < t_1 \leq \frac{C \times \vartheta}{\rho \times N_f} \tag{7}$$

Step 3 (Estimating the number of layers L) the number of layers L is based on the average traffic generated per frame in each layer which is given by the following equation:

$$\lambda_{avg} = n \times \lambda \times T_f \tag{8}$$

Then, the total active time should be greater than the time needed to send all the packet generated by the nodes:

$$L \times t_1 \geq \lambda_{avg} \times \left(\tau_t + \tau_p + 2\tau_d + \frac{W}{2}\tau_p\right) \tag{9}$$

Thus, L is bounded from below by:

$$L \geq \frac{\lambda_{avg} \times \left(\tau_t + \tau_p + 2\tau_d + \frac{W}{2}\tau_p\right)}{t_1} \tag{10}$$

Moreover, the dead time between layers t_2 is governed by the inequality:

$$t_2 > \tau_p + 2\tau_d \tag{11}$$

Therefore, the upper limit in L is:

$$L(t_1 + t_2) \leq T_f \tag{12}$$

Thus, L has the following design bounds:

$$\frac{\lambda_{avg} \times \left(\tau_t + \tau_p + 2\tau_d + \frac{W}{2}\tau_p\right)}{t_1} \leq L \leq \frac{T_f}{(t_1 + t_2)} \tag{13}$$

Other specifications and requirements in the application, such as delay limitations and buffer size in the node, can be used to determine the values of these timing parameters and to specify how many layers should be deployed to get the best performance.

IV. SIMULATIONS AND RESULTS

The performance of ML-MAC is simulated using MATLAB version 7.10.0.499 (R2010a) on a PC machine to compare the results with other MAC protocols that have been proposed for wireless sensor networks. The following subsection explains the assumptions that are made to implement ML-MAC as a MAC protocol for wireless sensor networks:

- A sensor node generates packets that follow the Poisson distribution.
- Time is divided into frames where each frame is Composed of listen and sleep periods.

TABLE I. MODEL PARAMETERS

Parameter	Value	Unit
Average packet inter-arrival time *T*	2-11	s
Number of access layers *L*	2-10	
Number of nodes *n*	100	
Frame duration	1	s
Layer duration t_1	0.3/L	s
Number of initial reservation slots *W*	8	
Node transmitting power	24.75	mW
Node listening power	13.5	mW
Node sleeping power	15	µW
Node transmission data rate	19.2	Kbps
Average packet length (α)	38	bytes
Simulation time	200	s

- Each node has three modes of operation: transmit, listen, and sleep.
- Nodes have unlimited transmit and receive buffer sizes.
- All MAC operations are based on the IEEE 802.1 1.
- The wireless channel is assumed to be perfect, i.e., there are no channel impairments.
- The radio transceiver of the node is TR1000 from RF Monolithic [12]. The power consumption for this radio transceiver in transmitting, listening, and sleeping modes are 24.75mW, 13.5mW and 15pW respectively. The transmission data rate of this radio transceiver is 19.2 Kbps.

According to assumption 4, packets will not be dropped as they are all ultimately going to be sent to their destinations. The parameter values chosen for numerical simulations are summarized in Table 1.

A. *Traffic Inter-arrival time Model*

In this simulation traffic model was chosen to be the Poisson distribution. The assumption of Poisson distribution for the traffic implies that nodes statistically generate traffic that is based on an exponentially distributed inter-arrival time [13]. This traffic model was chosen to test the protocol's performance for different arrival rates. Let the inter-arrival time between two successive packets be the random variable T. Then, the PDF (Probability Density Function) for the inter-arrival time of Poisson traffic follows the exponential distribution that can be expressed as:

$$f_T(t) = \lambda e^{-\lambda t} \qquad (14)$$

These parameters are [13]:

λ: Average Data rate

α : Average packet length in bits

σ : maximum burst rate.

Therefore, the inter-arrival time distribution is modified to get the shifted exponential distribution that can be expressed as [13]:

$$f_T(t) = be^{-(t-a)} \qquad \text{For } t \geq a \qquad (15)$$

Where $a > 0$ is the position parameter which represents the minimum time between adjacent packets and b is the shape parameter that determines how fast the exponential function decays with time.
The values of a and b for a source with parameters λ, σ and α, can be calculated as in [13] from the following equations:

$$a = \frac{\alpha}{\sigma} \qquad (16)$$

$$b = \frac{\sigma\lambda}{\alpha(\sigma-\lambda)} \qquad (17)$$

In this simulation, the average packet length α was assumed to be fixed with only 38 bytes since most wireless sensor networks have a very small packet size. Also, the average inter-arrival time *T* of the packets in this simulation was varied from 2 to 10 seconds. Therefore, λ and σ can be found based on the packet inter-arrival time from the following two equations:

$$\lambda = \frac{1}{T} \qquad (18)$$

$$\sigma = \frac{1}{T-\theta} \qquad (19)$$

Where θ is a constant value between 1 and (T - 1). In this simulation, θ was assumed to be 1.

To make the simulation simpler, the traffic is first generated at the beginning of the simulation for all the nodes in the networks for the entire simulation time. Each packet generated from any node is stored in the node transmit buffer and is assigned three flags: Arrival time, Destination node address and Reservation slot address These flags are used to calculate the time and the energy required to send that packet to its destination.

B. *Traffic destination Model*

The destination of each packet generated by a node is selected using the uniform random distribution for the non-coherent case where the destination could be any other node in any access layer including the same access layer. On the other hand, in the coherent case, the destination of a packet could be any other node in the same access layer which would give the best performance. Results are shown for both cases in order to test the performance of the protocol for different extreme situations.

C. Data Gathering

According to Table 4.1 time is divided into frames of 1s duration and the simulation time is 200s. The duty cycle is 33% which makes the duration of the listen period 300 ms for the S-MAC. However, for the ML-MAC with L layers, the listen period is 300/L ms.. The size of the data packet is fixed with 38 bytes which takes only 20ms to send in a typical radio channel [12]. In this simulation, the time index is set to be frame duration/1000, i.e., frames are divided into 1000 slots. The total energy consumed by each node over the entire simulation time is determined by calculating the time each node spends in the three modes, i.e., listen, transmit and sleep. Then, the total time nodes spend in each mode is multiplied by the amount of power consumed in that mode to get the total energy consumed by the node. Delay in this simulation is the sum of the time a packet may encounter in the transmit buffer and the time needed to send that packet. Therefore, the queuing delay is the dominant part that affects the delay. Delay is calculated by subtracting the time a packet is received by the destination from the time it was generated. A collision occurs where two or more nodes try to access the channel at exactly the same time index. The collided nodes have to back-off. The probability of collision is calculated by dividing the number of collisions by the total number of packets generated.

D. Protocol Performance

The main advantage of deploying multiple layers in ML-MAC is the reduction in node energy consumption as the following subsection illustrates. Another advantage of deploying multiple layers in ML-MAC is the reduction in the probability of collisions, which also saves energy that would otherwise have been needed for retransmitting the collided packets.

1) Overall energy consumption: Fig. 3 compares the average energy consumed by a node for IEEE 802.11, S-MAC and ML-MAC when L = 3. In this simulation, the non-coherent case is used. It shows that ML-MAC consumes 52% less energy than S-MAC when the traffic is heavy, i.e., the message inter- arrival time is less than about 5 seconds, and by 64% when the traffic is light, i.e., the message inter-arrival time is greater than about 5 seconds. In ML-MAC, the listen periods are shorter than S-MAC which results in this reduction of energy consumption. Fig. 4 shows the total energy consumed in a node for the whole simulation time, as the number of layers L changes from 1 (like S-MAC) to 10 layers using the non- coherent case. Traffic is generated with an average inter-arrival time T of 5s (λ = 0.2 packets/s). When the number of layers is less than five, the energy consumed decreases dramatically by adding more layers. However, after five layers, energy consumption will not be reduced that much since most of the packets will be destined to other layers and the nodes will spend more time waking up at different schedules. Also, this will increase the number of control packets that would consume more energy. The energy saving is inversely proportional to the number of layers deployed.

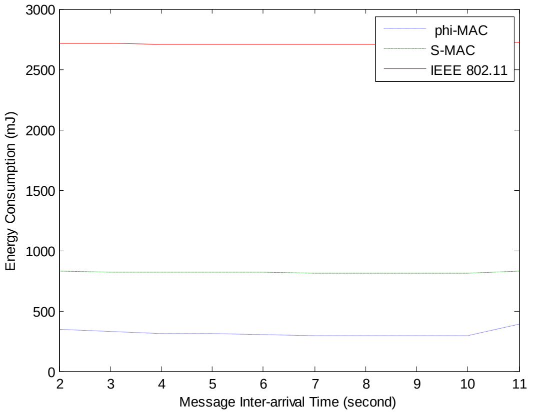

Fig. 3. Total energy consumption per node for IEEE 802.11, S-MAC and ML-MAC with L = 3; for the non-coherent case.

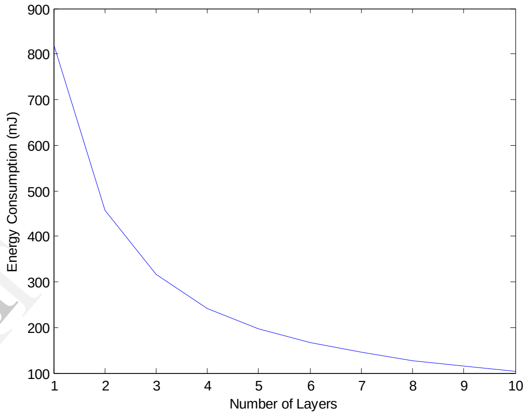

Fig. 4. Energy consumption per node for ML-MAC in the nun-coherent case, traffic is fixed: λ = 0.2 Packets/s.

2) Average packet delay: As the nodes sleep more in ML-MAC, packets will encounter more delay. This delay is the latency that a packet may encounter because it is stored in the node transmit buffer until it is sent successfully without a collision to its destination. Therefore, the delay here is composed of two components:

- Queuing delay because a packet could be destined to another layer or it has been generated while the node is in sleep mode.
- Transmission delay.

Therefore a packet will be delayed in ML-MAC by about one frame period. This result is shown in Fig. 5 where ML-MAC in the non-coherent case has a longer delay than S-MAC and IEEE 802.11. However, the increased delay in ML-MAC is not proportional to the number of layers as will be shown in the next result. The percentage of time when each node is in the sleep mode for S-MAC in this simulation is fixed at 70% because it has a fixed duty cycle. However, for the same duty cycle, nodes in ML-MAC sleep about 90% of the time and vary depending on the traffic type for the non-coherent case. The IEEE 802.11 has no sleep mode in this simulation which would result in minimum delay. Fig. 6 shows the effect of adding more layers on delay for the non-coherent case. If the number of layers is less than three, then delay would increase rapidly.

But, when we add more layers, then packets will not encounter more delay because they are usually buffered for the next or third frame cycle.

3) Probability of collision: Fig. 7 shows how the probability of collisions declines dramatically by adding more layers for ML-MAC using the non-coherent case and fixing the traffic at λ = 0.2 packets/s. However, after about 6 layers, it stops decreasing significantly because packet requests per layer spread out enough such that the chance of collision is reduced for this type of traffic. The high probability of collision in the last result is due to the traffic type generated for the simulation. The values of two traffic parameters λ and σ, described in section IV-A, are 0.2 and 0.25 packets/s, respectively. Because λ and σ are close to each other, then all 100 nodes generate packets that have around the same arrival times. As a result the probability of collision is high.

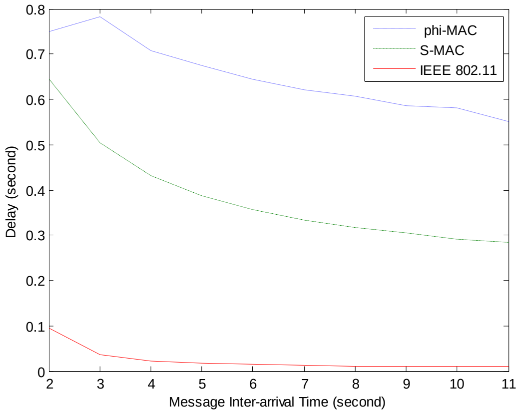

Fig. 5. Average delays for all packets sent for the three protocols: IEEE 802.1 1, S- MAC, and ML-MAC with L =3; for the non-coherent.

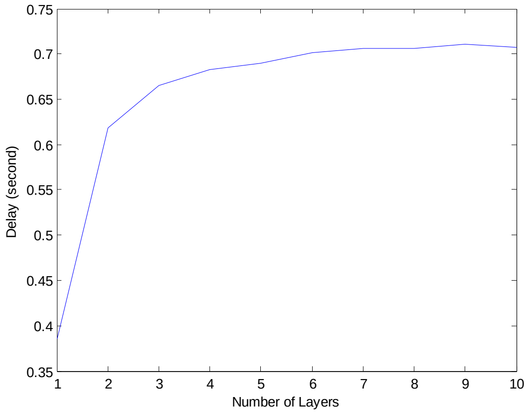

Fig. 6. Average delay for all packets sent for ML-MAC in the non-coherent case, traffic is fixed: λ = 0.2 Packets/s.

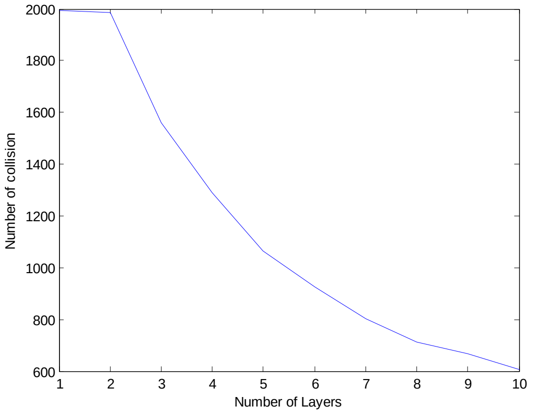

Fig. 7. Number of collisions for ML-MAC in the non-coherent case, traffic is fixed: $\boldsymbol{\lambda}$ =0.2packets/s.

V. CONCLUSIONS

ML-MAC was implemented and simulated for a simple network with 100 nodes that can communicate with each other directly, i.e., using a single one-hop network. The radio range of sensor nodes is very short, which covers only a small area. In order to be more practical by running the simulation on a multi-hop network environment that can cover a wide area, a routing protocol is required along with the MAC protocol to provide connectivity in the network. However, since ML-MAC is a modification to IEEE 802.11 and S-MAC, it is enough to be simulated in a simple network to show its efficiency and improvements. Therefore, the next step for ML-MAC is to implement it in a real hardware environment and test its efficiency as a MAC protocol for wireless sensor networks.

REFERENCES

[1] Ian F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine,* vol. 40, no. 8, pp. 102-1 16, Aug. 2002.

[2] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated, adaptive sleeping for wireless sensor networks," IEEE/ACM Transactions on Networking, vol. 12, no. 3, pp. 493-506, June 2004.

[3] Tao Zheng, Sridhar Radhakrishnan, Venkatesh Sarangan, "PMAC: An adaptive energy- efficient MAC protocol for Wireless Sensor Networks", Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05).

[4] L. Li and P. Sinha, "Throughput and energy efficiency in topology-controlled multi- hop wireless sensor networks," in WSNA103, San Diego, California, USA, Sept. 2003.

[5] C. S. R. Murthy and B. Manoj, Ad hoc wireless networks: architectures and protocols. Upper Saddle River, NJ, USA: Prentice Hall, 2004.

[6] R. Kannan, R. Kalidindi, S. Iyengar, and V. Kumar, "Energy and rate based MAC protocol for wireless sensor networks," SIGMOD Record, vol. 32, no. 4, Dec. 2003.

[7] Liansheng Tan, Jintao Meng, Jie Li, Han-Chieh Chao, "PH-MAC: A Periodically Hybrid MAC Protocol for Wireless Sensor Networks," Journal of Internet Technology vol. 8, no. 4, 2007.

[8] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.1 1, 1997.

[9] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in Proceedings of the 21st International Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002), vol. 3, New York, NY, USA, June 2002, pp. 1567-1576.

[10] T. van Dam and K. Langendoen, "Energy-efficient MAC: An adaptive energy-efficient MAC protocol for wireless sensor networks," in Proceedings of the first international conference on embedded networked sensor systems (SenSys), Los Angeles, California, USA, Nov. 2003.

[11] A. Woo and D. Culler, "A transmission control scheme for media access in sensor networks," in ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom), Rome, Italy, July 200 1, pp. 221-235.

[12] "ASH transceiver TR1000 data sheet,"RF Monolithics Inc. [Online]. Available: http://www.rfm.com/.

[13] F. Gebali, Computer Communication Networks: Analysis and Design, 3rd Ed. Victoria, B.C, Canada: Northstar Digital Design, Inc., 2005.