# Delay and Energy Models for Polling Based MAC Protocols with Sleep-Wake Cycles

Haiming Yang and Biplab Sikdar
Department of ECSE, Rensselaer Polytechnic Institute, Troy, NY 12180 USA

***Abstract*—While decentralized medium access control (MAC) protocols are more popular in wireless networks, cluster based sensor networks are particularly amenable to centralized, polling based protocols. This paper presents an analytic model to evaluate the performance of a polling based MAC protocol in terms of the packet delay, buffer overflow rates and energy consumption. We show that the polling based protocol can outperform popular decentralized MAC protocols. Simulation results are presented to validate our model and conclusions.**

## I. Introduction

A key constraint on the design of MAC protocols for wireless sensor networks (WSNs) is their limited on board energy. Since the energy consumed by the nodes in idle listening of the channel causes significant battery drain, most MAC protocols for WSNs propose that nodes turn off their radios when not involved in ongoing transmissions. Decentralized contention based MAC protocols that use different variants of sleep-wake cycles have been studied extensively in literature [1], [2]. The performance of these decentralized protocols, however, degrades as the network load increases due to the increased incidence of collisions and the associated bandwidth wastage. Cluster based WSN architectures [3], on the other hand, are particularly suitable for centralized, polling based MAC protocols though their performance with sleep-wake cycles has not been previously explored.

In this paper we develop analytic models to evaluate the performance of a polling based MAC protocol with sleep-wake cycles for WSNs. We first develop a queueing model to evaluate the average packet delays and then use the results to evaluate the per node energy consumption rates. The proposed model is also used to evaluate the packet loss rates due to buffer overflow at the nodes. Simulation results are then used to validate the analysis and also to demonstrate the superior performance of the polling based MAC protocol with sleep-wake cycles over similar decentralized protocols.

The rest of the paper is organized as follows. Section II presents the queueing model and Section III presents the energy consumption model. Section IV presents the simulation results and comparison with decentralized protocols. Finally, Section V concludes the paper.

## II. Delay Model

### A. Protocol Description

We assume a cluster based WSN architecture wherein sensors in a geographical region select a node amongst them as the cluster head. The cluster head is responsible for communicating with other cluster heads and the sink. All other nodes are leaf nodes, and can only communicate with the cluster head in their cluster.

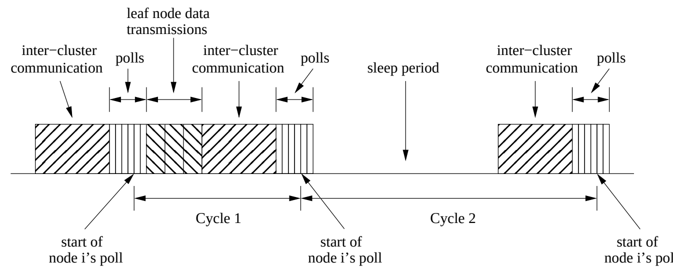


Fig. 1. Protocol operation showing two cycles of transmission. Cycle 1 has three data transmissions while Cycle 2 has none, resulting in a sleep period.

The MAC protocol's data exchange process is divided into rounds. A round begins with the inter-cluster period where cluster heads exchange data with other cluster heads or with the sink, and leaf nodes may turn off their radios. It is followed by the intra-cluster period where cluster heads exchange data with their leaf nodes. The polling based MAC protocol applies to the intra-cluster communication. In the intra-cluster period, the cluster head first polls *all* its leaf nodes and then assigns them time slots to transfer their data. Only nodes with data are assigned slots and the remaining nodes may sleep till the end of the round. The intra-cluster period ends when all nodes have been polled in a round. Also, if none of the nodes have any data to send when polled, the cluster transitions into a sleep state where all leaf nodes turn off their radios. A new round starts when the sleep period ends.

### B. Queueing Model

Consider an arbitrary cluster with $M$ nodes. Each node is assumed to have $K$ buffers to store packets. Let the channel rate be $\frac{1}{C}$ bytes/second and the data packets generated by each node be of $k_D$ bytes, requiring $T_D = k_D C$ seconds to be transmitted. During each poll, the cluster head transmits $k_{P\_dl}$ bytes to the polled node and the polled node replies using $k_{P\_ul}$ bytes. We use the notation $k_P = k_{P\_dl} + k_{P\_ul}$ and the time to poll a node is $T_P = k_P C$ seconds.

The packet interarrival times at each node are assumed to distributed according to a Markov modulated Poisson process (MMPP) with an arbitrary number of states, $r$. An MMPP based arrival process is used in this paper because of their versatility in modeling traffic types such as voice, video as well as long range dependent traffic [4]. The MMPP is characterized by the transition rate matrix $\mathbf{R}$ and the diagonal rate matrix $\mathbf{\Lambda}$ that contains the arrival rates at each state:

$$\mathbf{R} = \begin{bmatrix} -\sigma_1 & \sigma_{12} & \cdots & \sigma_{1r} \\ \sigma_{21} & -\sigma_2 & \cdots & \sigma_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{r1} & \sigma_{r2} & \cdots & -\sigma_r \end{bmatrix}, \mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_r \end{bmatrix} \quad (1)$$

The steady state probability vector $q$ of the Markov chain satisfies $q\mathbf{R} = 0$ and $qe = 1$ where $e$ is a unit vector. The average arrival rate at a node is then given by $\lambda = q\mathbf{\Lambda}$.

For our analysis, we consider the system operation from leaf node $i$'s perspective to be divided into periods of variable length called *cycles*. A cycle begins when the polling of node $i$ starts, and ends when node $i$ is polled the next time. The duration of a cycle is denoted by $T_C$. Figure 1 shows the operation of the MAC protocol for two cycles. Note that the probability distribution of the duration of a cycle is identical to that of a round (defined earlier).

Our analysis is based on modeling the MAC layer behavior of each leaf node as a MMPP/G/1/K queue. Our first step is to characterize the service time distribution. There are $M-1$ leaf nodes in a $M$ node cluster. The data arrival rate in the cluster is thus $(M-1)\lambda$. Consider a tagged packet arriving at leaf node $i$, $1 \leq i \leq M-1$. At the instant of its arrival, the queue at the leaf node may be in one of two states: **1. S0:** The queue is empty **2. S1:** The queue is non-empty. Next we consider the service time for these two cases.

*1) Arrival at an Empty Queue: State S0:* Consider the cycle in which the tagged packet arrives. The queue at node $i$ is empty when the packet arrives but may not have been so at the beginning of the cycle when it was polled. Thus we consider two subcases corresponding to whether the queue at leaf node $i$ was empty (case C1) or not (case C2) when it was polled.

In case C1, since node $i$ was empty when it was polled in the current cycle, it cannot transmit any data in this cycle. Among the remaining $M-2$ leaf nodes, let $k$ be the number of nodes that transmit data in this cycle. We first consider the case where $k > 0$. Since each cycle also includes an inter-cluster communication period (of duration $T_I$) and the time to poll the $M-1$ leaf nodes, the length of the cycle, $T_C$, is given by $T_C = T_I + (M-1)T_P + kT_D$. For arbitrary arrivals independent of the departure process in a frame based departure system, an arrival is equally likely to occur anywhere in a frame [7]. In our case, *given* that an arrival occurs in a cycle, the arrival instance, $t$, *relative* to the start of the cycle is thus uniformly distributed in $[0, T_C]$, denoted by $U[0, T_I + (M-1)T_P + kT_D]$. The time the tagged arrival has to wait till the start of the next frame is $T_C - t$ the PDF of $T_C - t$ is also $U[0, T_I + (M-1)T_P + kT_D]$. In the next cycle, the tagged packet first has to wait for the polls of $M-i$ leaf nodes, including itself. If $j$ of the $i-1$ leaf nodes polled before node $i$ also have data to transmit when they are polled, the tagged packet has to wait for an additional $jT_D$ seconds before it is served. Let $\rho$ denote the probability that the queue at any leaf node is empty at an arbitrary time instant. The probability that $j$ of the $i-1$ leaf nodes had non-empty queues when they were polled in the current cycle and thus transmit data in this cycle is binomially distributed with parameters $B[i-1, 1-\rho]$. The Laplace-Stieltjes Transform (LST) of the service time in this case, $X_{i,k,S0,C1}$, is

$$\begin{aligned} H_{X_{i,k,S0,C1}}(s) &= \text{LST}\,[U[0, T_I + (M-1)T_P + kT_D] + \\ &\qquad (M-i)T_P + B[i-1, 1-\rho]T_D + T_D] \quad (2) \\ &= \frac{1 - e^{-s(T_I + (M-1)T_P + kT_D)}}{s(T_I + (M-1)T_P + kT_D)} \frac{(\rho + (1-\rho)e^{-sT_D})^{i-1}}{e^{s((M-i)T_P + T_D)}} \quad (3) \end{aligned}$$

where the first term in the equation above is the LST of $U[0, T_I + (M-1)T_P + kT_D]$, the denominator of the second term is the LST of the constants $(M-i)T_P + T_D$ and the numerator of the second term is the LST of $B[i-1, 1-\rho]L$.

To uncondition Eqn. (3) on $k$, we note that the number of active nodes among $M-2$ leaf nodes is binomially distributed as $B[M-2, 1-\rho]$. In the special case where $k = 0$, none of the leaf nodes have any data to send when they are polled in the current cycle. Thus the nodes enter the sleep period and the length of the cycle is $T_C = T_I + (M-1)T_P + T_S$, where $T_S$ is the duration of a sleep period. Once the cycle ends, as for the case with $k > 0$, the tagged packet at leaf node $i$ first waits for the polls of $M-i$ leaf nodes and then for an additional $jT_D$ seconds for data transmissions where $j$ is binomially distributed as $B[i-1, 1-\rho]$. Then, unconditioning Eqn. (3) on $k$ and adding to it the case for $k = 0$, the LST of the service time for case C1, $X_{i,S0,C1}$, is given by

$$\begin{aligned} H_{X_{i,S0,C1}}(s) = \frac{(\rho + (1-\rho)e^{-sT_D})^{i-1}}{e^{s((M-i)T_P + T_D)}} \Bigg[ \rho^{M-2} \frac{1 - e^{-s(T_I + (M-1)T_P + T_S)}}{s(T_I + (M-1)T_P + T_S)} \\ + \sum_{k=1}^{M-2} \binom{M-2}{k} (1-\rho)^k \rho^{M-2-k} \frac{1 - e^{-s(T_I + (M-1)T_P + kT_D)}}{s(T_I + (M-1)T_P + kT_D)} \Bigg] \end{aligned}$$

In case C2, node $i$ was non-empty when it was polled but was empty when the tagged packet arrived in the same cycle. Thus, the tagged packet must have arrived after leaf node $i$ transmitted its data in the current cycle. The remaining time in the cycle after the packet from leaf node $i$ is transmitted is $kT_D + T_I + (i-1)T_P$, if $k$ of the $M-1-i$ nodes polled after node $i$ also transmit data in the current cycle. Now, the tagged packet has to wait for $T_C - t$ seconds for the current cycle to end. The PDF of $t$, given that the tagged packet arrived after node $i$ transmitted its data in the cycle is distributed as $U[T_C - kT_D - T_I - (i-1)T_P, T_C]$. Thus $T_C - t$ is distributed as $U[0, kT_D + T_I + (i-1)T_P]$. In the next cycle, the tagged packet first waits for the polls of $M-i$ leaf nodes, including itself, and transmissions from $j$ of the $i-1$ leaf nodes polled before node $i$. As before, $j$ is binomially distributed with parameters $B[i-1, 1-\rho]$. The LST of the service time in this case, $X_{i,k,S0,C2}$, is given by

$$\begin{aligned} H_{X_{i,k,S0,C2}}(s) &= \text{LST}\,[U[0, kT_D + T_I + (i-1)T_P] + \\ &\qquad (M-i)T_P + B[i-1, 1-\rho]T_D + T_D] \\ &= \frac{1 - e^{-s(kT_D + T_I + (i-1)T_P)}}{s(kT_D + T_I + (i-1)T_P)} \frac{(\rho + (1-\rho)e^{-sT_D})^{i-1}}{e^{s((M-i)T_P + T_D)}} \quad (4) \end{aligned}$$

To uncondition Eqn. (4) on $k$, we use the fact that the number of active nodes among $M-1-i$ leaf nodes is binomially distributed as $B[M-1-i, 1-\rho]$. Then, the LST of the service time for case C2, $X_{i,S0,C2}$, is

$$H_{X_{i,S0,C2}}(s) = \sum_{k=0}^{M-1-i} \binom{M-1-i}{k} (1-\rho)^k \rho^{M-1-i-k} \frac{1-e^{-s(kT_D+T_I+(i-1)T_P)}}{s(kT_D+T_I+(i-1)T_P)} \frac{(\rho+(1-\rho)e^{-sT_D})^{i-1}}{e^{s((M-i)T_P+T_D)}} \quad (5)$$

The probabilities of cases C1 and C2 are $P[C1] = \rho$ and $P[C2] = 1-\rho$, respectively. Combining cases C1 and C2, the LST of the service time in state S0, $X_{i,S0}$, is then given by

$$\begin{aligned} H_{X_{i,S0}}(s) &= \rho H_{X_{i,S0,C1}}(s) + (1-\rho) H_{X_{i,S0,C2}}(s) \\ &= \frac{(\rho+(1-\rho)e^{-sT_D})^{i-1}}{e^{s((M-i)T_P+T_D)}} \Bigg[ \sum_{k=0}^{M-1-i} \binom{M-1-i}{k} (1-\rho)^{k+1} \rho^{M-1-i-k} \\ &\quad \frac{1-e^{-s(kT_D+T_I+(i-1)T_P)}}{s(kT_D+T_I+(i-1)T_P)} + \rho^{M-1} \frac{1-e^{-s(T_I+(M-1)T_P+T_S)}}{s(T_I+(M-1)T_P+T_S)} \\ &\quad + \sum_{k=1}^{M-2} \binom{M-2}{k} (1-\rho)^k \rho^{M-1-k} \frac{1-e^{-s(T_I+(M-1)T_P+kT_D)}}{s(T_I+(M-1)T_P+kT_D)} \Bigg] \end{aligned} \quad (6)$$

*2) Arrival at a Non-Empty Queue: State S1:* For these arrivals, the service time starts when the last of the enqueued packets departs the queue. Once the tagged packet comes to the head of the queue, it first has to wait for the current cycle to finish. In the remainder of the current cycle, any of the remaining $M-1-i$ nodes may transmit their data, and we also have an inter-cluster communication period and the polls of $i-1$ leaf nodes. Before the tagged packet receives service in the next cycle, we have $M-i$ polls including the poll of leaf node $i$, along with possible data transmissions from the $i-1$ leaf nodes polled before leaf node $i$. Since the number of nodes with data transmissions in a cycle is binomially distributed, the LST of the service time for arrivals in state S1, $X_{i,S1}$, is

$$\begin{aligned} H_{X_{i,S1}}(s) &= \text{LST}\,[B[M-1-i, 1-\rho]T_D + T_I + (M-1)T_P \\ &\quad + B[i-1, 1-\rho]T_D + T_D] \\ &= e^{-s(T_I+(M-1)T_P+T_D)} (\rho+(1-\rho)e^{-sT_D})^{M-2} \end{aligned} \quad (7)$$

*3) Overall Service Time, Delay Distribution and Loss Rates:* Combining the cases S0 and S1, the LST of the service time of an arbitrary arrival at SS $i$, $X_i$, is given by

$$H_{X_i}(s) = \rho H_{X_{i,S0}}(s) + (1-\rho) H_{X_{i,S1}}(s) \quad (8)$$

where $H_{X_{i,S0}}(s)$ and $H_{X_{i,S1}}(s)$ are given in Eqn. (6) and Eqn. (7) respectively. The average service time, $\Theta$, is denoted by $\Theta = -\frac{d}{ds} H_{X_{i,S1}}(s)\big|_{s=0}$ and given by

$$\begin{aligned} \Theta &= \frac{2M-i-1}{2} T_P + \frac{T_I}{2} + \left[ 1 + \frac{(M-3+i)(1-\rho)}{2} \right] T_D \\ &\quad + (1-\rho) \left[ \frac{(i-1)T_P + T_I + (M-1-i)(1-\rho)T_D}{2} \right] \\ &\quad + \rho \left[ \rho^{M-1} \frac{T_S}{2} + \rho \left[ \frac{(M-i)T_P + (i-1)(1-\rho)T_D}{2} \right] \right] \end{aligned}$$

To obtain the distribution of the packet delays and loss rates, the queue at each SS is modeled as a MMPP/G/1/K queue whose service time distribution is given by Eqn. (8). We use the analysis for the MMPP/G/1/K queue from [5] and list the equations below for completeness.

Consider the imbedded Markov chain consisting of the service completion instants at the queue. Let $\pi(k)$ (respectively, $p(k)$) be the $r-$dimensional vector whose $j-$th element is the limiting probability at the imbedded epochs (respectively, at an arbitrary time instant) of having $k$ packets in the queue and being in the phase $j$ of the MMPP, $k = 0, 1, \cdots, K-1$ (respectively, $k = 0, 1, \cdots, K$). Consider the matrix sequence $\{\mathbf{C}_k\}$ defined as

$$\mathbf{C}_{k+1} = \left[ \mathbf{C}_k - \mathbf{U}\mathbf{A}_k - \sum_{\nu=1}^{k} \mathbf{C}_\nu \mathbf{A}_{k-\nu+1} \right] \mathbf{A}_0^{-1} \quad (9)$$

for $k = 1, 2, \cdots, K-2$ with $\mathbf{C}_0 = \mathbf{I}$, $\mathbf{C}_1 = (\mathbf{I} - \mathbf{U}\mathbf{A}_0)\mathbf{A}_0^{-1}$ and $\mathbf{I}$ being a $r \times r$ identity matrix. The $(k,l)-$th element of the matrix $\mathbf{A}_\nu$ denotes the conditional probability of reaching phase $l$ and having $\nu$ arrivals at the end of a service time, starting from phase $k$. The matrices $\mathbf{A}_\nu$ can be easily calculated using an iterative procedure [6]. The probability vectors $\pi(k)$ can then be calculated using

$$\pi(0) \left[ \sum_{\nu=0}^{K-1} \mathbf{C}_\nu + (\mathbf{I} - \mathbf{U})\mathbf{A}(\mathbf{I} - \mathbf{A} + eq)^{-1} \right] = q \quad (10)$$

and $\pi(k) = \pi(0)\mathbf{C}_k$, $k = 1, 2, \cdots, K-1$. The vectors $p(k)$ are then obtained using $p(0) = \xi \pi(0)(\mathbf{\Lambda} - \mathbf{R})^{-1}\Theta^{-1}$ and

$$p(k) = \xi \left[ \pi(k) + \sum_{\nu=0}^{k-1} \pi(\nu) \mathbf{U}^{k-1-\nu} (\mathbf{U} - \mathbf{I}) \right] (\mathbf{\Lambda} - \mathbf{R})^{-1} \Theta^{-1} \quad (11)$$

for $k = 1, 2, \cdots, K-1$ and $p(K) = q - \sum_{\nu=1}^{K-1} p(\nu)$ where $\xi = [1 + \pi(0)(\mathbf{\Lambda} - \mathbf{R})^{-1}\Theta^{-1}e]^{-1}$. The packet blocking probability is given by

$$P_b = 1 - \sum_{\nu=0}^{K-1} p(\nu) \quad (12)$$

Finally, the LST of the cumulative distribution function of the packet waiting time, $W(s)$ is given by

$$W(s) = \frac{1}{1-P_b} \left[ p(0) + \xi \Theta^{-1} \sum_{\nu=1}^{K-1} G_\nu(s) H_{X_i}^{K-1-\nu}(s) \mathbf{T}_{K-1-\nu}(s) \right] \quad (13)$$

where $G_j(s) = \pi(0)[\mathbf{I} - \mathbf{U}H_{X_i}(s)] - H_{X_i}^j(s)\pi(j)$, $\mathbf{T}_j(s) = \mathbf{F}(s)[-\mathbf{\Lambda}\mathbf{F}(s)]^j$ and $\mathbf{F}(s) = [s\mathbf{I} + \mathbf{R} - \mathbf{\Lambda}]^{-1}$. Moments of the packet waiting time can be easily obtained from Eqn. (13).

To complete the analysis, we note that the probability that the queue is empty at an arbitrary instant of time, $\rho$, is given by $\rho = p(0)e$. However, $\rho$ is used in the expressions for the service time, which are in turn used to evaluate $p(0)$. To obtain $\rho$, we use an iterative technique. Under this iterative strategy, we start with an arbitrary value of $\rho$ in $(0, 1)$ and use it to compute the service time distribution and $p(0)$. The new value of $\rho$ given by $\rho = p(0)e$ is then used to recalculate the service time distribution which is then used to find the new $p(0)$. This process continues till the values of $\rho$ and $p(0)e$ converge.

## III. Energy Consumption Model

The energy consumption of the MAC protocol depends on the time spent by each node in transmitting, receiving or in the sleep period, in addition to the energy dissipation characteristics of the radios used by the nodes. We assume that the radio dissipates $E_{elec}$ Joules/bit (J/bit) to run the transmitter or receiver circuitry and $E_{amp}$ J/m$^2$ for the transmitter amplifier to achieve an acceptable signal to noise ratio [3]. Assuming $d^2$ energy loss in the channel, to send a $k$ bits message to a distance $d$, the radio expends

$$E_{Tx}(k,d) = kE_{elec} + kd^2E_{amp} \tag{14}$$

and to receive this message, the radio expends

$$E_{Rx}(k,d) = kE_{elec} \tag{15}$$

To obtain the rate of energy consumption for the polling based MAC protocol, we first evaluate the average cycle time. Given that there are $k$ nodes with data transmissions in a cycle, the cycle length is $T_C = T_I + (M-1)T_P + kT_D$. Since $k$ is binomially distributed as $B[M-1, 1-\rho]$, the expected cycle length given that at least one node sends data is

$$E[T_C|k>0] = T_I + (M-1)T_P + \frac{(M-1)(1-\rho)}{1-\rho^{M-1}}T_D \tag{16}$$

In case of a sleep cycle, the cycle length is $T_C = T_I + (M-1)T_P + T_S$. Thus the expected cycle length is given by

$$E[T_C] = T_I + (M-1)T_P + \rho^{M-1}T_S + (M-1)(1-\rho)T_D \tag{17}$$

At any instant, a leaf node may be either in the polling, data transmission, inter-cluster or sleep period. During the polls in a cycle, each leaf node spends $k_{P\_ul}(E_{elec} + E_{amp}d^2) + k_{P\_dl}E_{elec}$ J of energy on its own poll and $(M-2)k_PE_{elec}$ J listening to the polls of other nodes. Since polls occur every cycle, each leaf node expends this amount of energy every $E[T_C]$ seconds. Also, the rate at which packets are accepted in the queue of each node is $\lambda(1-P_b)$ where the packet blocking probability, $P_b$, is given by Eqn. (12). In a stable system, the rate at which packets depart is thus also $\lambda(1-P_b)$. For each packet transmitted, a leaf node expends $E_{elec}k_D + E_{amp}k_Dd^2$ J of energy. Finally, we note that a node does not expend any energy during the inter-cluster and sleep periods. Using the expression for $E[T_C]$ from Eqn. (17), the total rate at which a leaf node spends energy is given by

$$E_{avg} = \frac{(M-1)k_PE_{elec} + k_{P\_ul}E_{amp}d^2}{T_I + (M-1)T_P + \rho^{M-1}T_S + (M-1)(1-\rho)T_D} + \lambda(1-P_b)(E_{elec}k_D + E_{amp}k_Dd^2) \tag{18}$$

## IV. Simulation Results

We implemented the polling based MAC protocol in the NS-2 simulator and in this section we use simulation results to verify our analysis and compare them against the performance of decentralized protocols. The length of each simulation run is 2000 seconds, and each result is the average of 20 runs. The channel data rate is 20Kbps, $T_P = 0.004$ sec, $T_I = 0.4$ sec and $T_D = 0.0256$ sec. There are 9 leaf nodes in each cluster. A 2-state MMPP with transition rates of $\sigma_{12} = 3.15$ and $\sigma_{21} = 1.94$ and the ratio $\lambda_1 = 1.6\lambda_2$ was used [4] for the arrival process. The radio parameters were assumed to be $E_{elec} = 50nJ/bit$ and $E_{amp} = 100pJ/m^2$ [3].

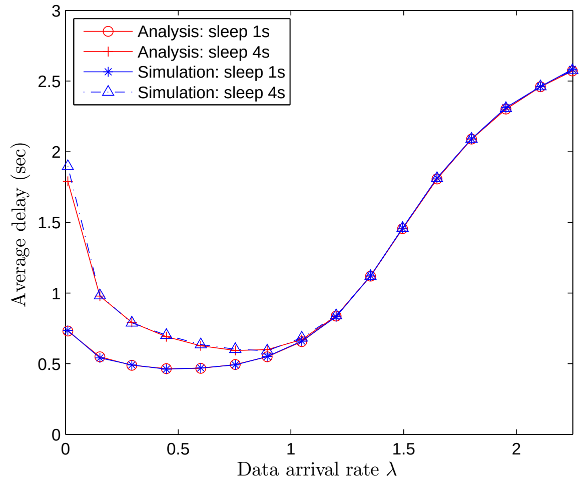


Fig. 2. Average packet delay.

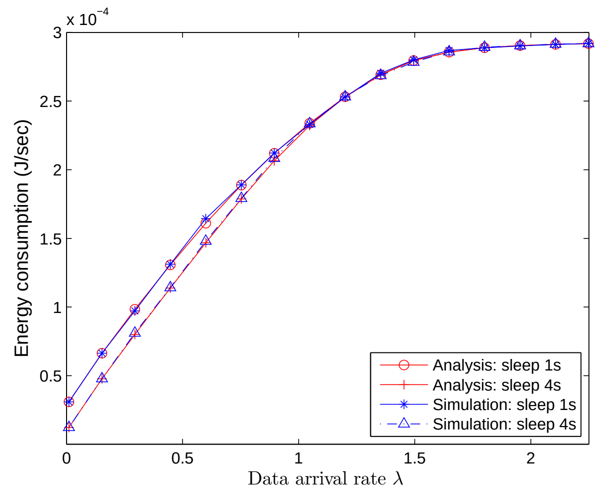


Fig. 3. Average rate of energy consumption.

In Figures 2 and 3, we compare the analytic and simulation results for the average packet delay and the rate of energy consumption at a leaf node 5 for different traffic loads and sleep periods of 1 and 4 seconds. We note the close match between the simulation and analytic results. Figure 2 shows that the minimum delay is not achieved at low arrival rates but at moderate loads. For low data rates, a large fraction of the arrivals occur when the system is in the sleep state. These arrivals need to wait for the relatively large sleep period to finish before they can be transmitted. As the arrival rate increases, the probability that an arrival occurs in a sleep period decreases, thereby reducing the delays incurred while waiting for the sleep period to finish. At high arrival rates, the queuing delay becomes dominant and the packet delay increases again. Consequently, there exists an unique arrival

| Data arrival rate $\lambda$ | Packet Loss Probability | | | | | |
|---|---|---|---|---|---|---|
| | $K=1$ | | $K=5$ | | $K=10$ | |
| | Ana. | Sim. | Ana. | Sim. | Ana. | Sim. |
| 1.0479 | 0.2895 | 0.3112 | 0.0021 | 0.0031 | 0.0000 | 0.0000 |
| 1.3535 | 0.3527 | 0.3622 | 0.0258 | 0.0283 | 0.0028 | 0.0022 |
| 1.6482 | 0.4090 | 0.4101 | 0.1171 | 0.1242 | 0.0943 | 0.0912 |
| 1.9539 | 0.4612 | 0.4842 | 0.2344 | 0.2456 | 0.2276 | 0.2471 |
| 1.9539 | 0.5053 | 0.5153 | 0.3294 | 0.3314 | 0.3274 | 0.3214 |

TABLE I
PACKET LOSS RATES FOR THE POLLING BASED MAC PROTOCOL FOR BUFFER SIZES OF 1, 5 AND 10.

rate that achieves the minimum delay, typically at moderate loads, where the total contribution of the delays from the sleep time and the queuing effect is lowest.

From Figure 3, we observe that increasing the sleep time reduces the energy consumption. However, this decrease cannot continue unboundedly because as the sleep time becomes longer, each sleep period will have a larger number of packet arrivals. These arrivals will queue up and consequently, the subsequent active periods also become longer. Also, the energy consumption for all sleep periods converges to a constant value as the load increases. This is because at high loads, the system does not enter the sleep period and each node almost always transmits a packet in a cycle. Thus cycle lengths are almost constant and each the energy consumption rates saturate.

To evaluate the packet loss probabilities, Table I shows the analytic and simulation results for buffer sizes of 1, 5 and 10 for a sleep period of 1 second. Again, we note that the results match closely.

Finally, we compare the performance of the polling based MAC protocol with the popular decentralized protocol with sleep-wake cycles: SMAC [1]. We compare the performance at low data rates since the contention based SMAC has high collision rates at high traffic loads and its performance degrades. To compare the protocol performance in similar settings, parameters were selected such that the packet delays of the protocols are similar. The length of each simulation run was 8000 seconds and the channel data rate was 2Mbps. We used SMAC with a duty cycle of 10% and $T_P = 0.00004$ sec, $T_I = 0.4$ sec, $T_S = 1$ sec, and $T_D = 0.000256$ sec for the polling based scheme. The results are shown in Figures 4 and 5. The polling based scheme outperforms SMAC in terms of the delay as well as the energy consumption. Interestingly, the polling based scheme has at least 100% lower energy consumption as compared to SMAC and the difference is larger at higher loads. This is because SMAC: (1) uses a fixed sleep-wake schedule and does not adapt to the changing traffic conditions, resulting in energy wastage and (2) wastes energy through the collisions resulting from its contention based MAC protocol.

## V. CONCLUSION

This paper presents analytic models to evaluate the delay, loss rates and energy consumption characteristics of a polling based MAC protocol with sleep-wake cycles for WSNs. The performance of polling based MAC protocols is compared against similar decentralized protocols and is shown to have superior performance in terms of both delay and energy.

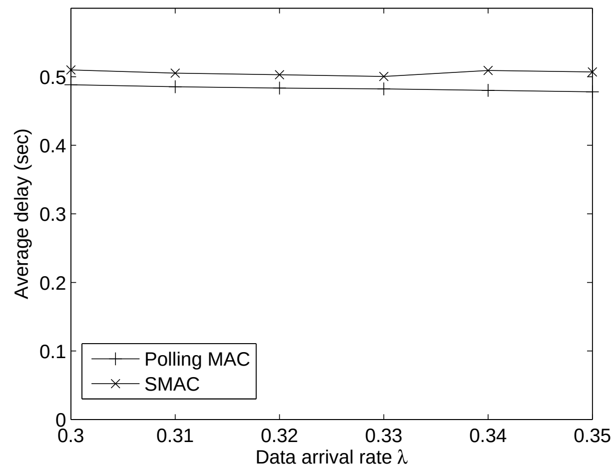


Fig. 4. Comparison of packet delays between SMAC and polling based MAC protocols.

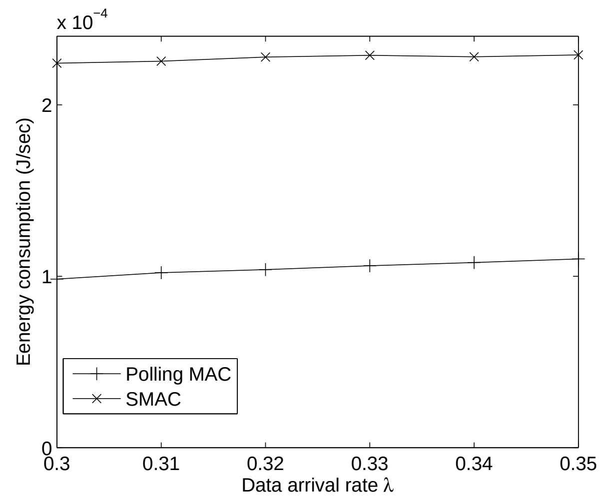


Fig. 5. Comparison of energy consumption between SMAC and polling based MAC protocols.

## REFERENCES

[1] 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.
[2] S. Du, A. Saha and D. Johnson, "RMAC: A Routing-Enhanced Duty-Cycle MAC Protocol for Wireless Sensor Networks," *Proc. IEEE INFOCOM*, pp. 1478-1486, Anchorage, AK, May 2007.
[3] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660-670, October 2002.
[4] S. Shah-Heydari and T. Le-Ngoc, "MMPP models for multimedia traffic," *Telecommunication Systems*, vol. 15, no. 3-4, pp. 273-293, December 2000.
[5] A. Baiocchi and N. Blefari-Melazzi, "Steady-state analysis of the MMPP/G/1/K queue," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 531-534, April 1993.
[6] W. Fischer and K. Meier-Hellstern, "The Markov-modulated Poisson-process (MMPP) cookbook," *Performance Evaluation*, vol. 18, no. 2, pp. 149-171, September 1993.
[7] D. Bersekas and R. Gallager, *Data Networks*, Prentice Hall, NJ, 1987.