Proceedings of the 45th IEEE Conference on Decision & Control
Manchester Grand Hyatt Hotel
San Diego, CA, USA, December 13-15, 2006

Dynamic Sleep Time Control in Event-Driven Wireless Sensor Networks

Xu Ning and **Christos G. Cassandras**
Dept. of Manufacturing Engineering and Center for Information and Systems Engineering
Boston University, Boston, MA 02215
nx@bu.edu, cgc@bu.edu

Abstract— **One of the main sources of energy waste in Wireless Sensor Networks (WSN) is idle listening, i.e., nodes consuming energy listening to an idle channel. In this paper, we present a dynamic sleep time control method exploiting known traffic statistics to sample the channel more frequently when it is likely to have traffic and less frequently when it is not. When such information is not a priori available, we present an iterative algorithm to learn the statistics and adapt the sleep time control policy as time evolves. Simulation results are included to compare fixed sleep times to a dynamic control policy.**

I. Introduction

A Wireless Sensor Network (WSN) consists of low-cost nodes which are mainly battery powered and have sensing and wireless communication capabilities [1]. Usually, the nodes in such a network share a common objective such as environmental monitoring or event detection. Power consumption is a key issue in WSNs, since it directly impacts their lifespan in the likely absence of human intervention for most applications of interest.

Energy in WSN nodes is consumed by the CPU, by sensors/actuators, and by radio, with the latter consuming the most. In [2] it is reported that *idle listening*, which occurs when a radio receiver remains listening during an idle period in the network, consumes the most energy required for receiving. Various approaches for reducing idle listening have been investigated. These approaches can be categorized according to the network layer they apply to [3], including the application layer (TinyDB [4], etc), network layer (Flexible Power Scheduling [3], etc), MAC layer (S-MAC [2], T-MAC [5], etc) and lastly the physical layer: low-power listening/preamble sampling (i.e., WiseMAC [6]).

The key idea in low-power listening/preamble sampling techniques is that each message at the sender side is attached to a "preamble" of some length. The receiver periodically turns on for a very short time to detect whether the channel is busy. If there is traffic, the receiver will remain on and receive the message. Since in WSNs traffic is usually sporadic, most of the time the channel is idle. Using preamble sampling techniques, the duty cycle of the radio can be reduced to very low levels during the channel's idle time. Such techniques have been adopted, for instance, in radio paging systems [7]. Preamble sampling with non-persistent CSMA protocols have been proposed in [8] specifically for sporadic control/signaling traffic in WSNs. WiseMAC [6] further optimizes the scheme by reducing the preamble length according to the message interval and clock drift. B-MAC [9] also uses a low-power listening technique where the detailed energy cost of each sample is explicitly measured.

Many of the above approaches have one element in common: periodic sleep time control. Depending on the tasks they perform, WSNs can be differentiated between continuous monitoring and event driven WSNs [10], in which network activities are triggered by external random events. Periodic sleep time control can still apply in event driven WSN. However, since the event times are not deterministic, many wake-ups actually take place during times when an event is *not likely* to happen. An obvious question therefore arises: if we have some a priori information about the event, can we achieve better results?

The contribution of this paper is to present a dynamic sleep time control approach based on the idea of selecting sleep time intervals (i.e., sampling periods) using statistical information available as well as prior observations. We show that when a desired expected delay in transmitting a message is specified, a sleep time interval can be evaluated depending on the message interarrival time distribution. Moreover, if this distribution is unknown, we further propose an adaptive scheme using a stochastic approximation algorithm.

The dynamic sleep time control problem is described in Section II, where the key result on which the proposed method is based is also derived. In Section III, this result is applied to some known interarrival time distributions, including the interesting special case of the exponential distribution. In Section IV, we develop an adaptive approach in order to handle the case of unknown distributions and adjust sleep times on line. Simulation results are provided in Section V and conclusions are given in Section VI.

II. Dynamic Sleep Time Control

We consider a typical WSN configuration widely adopted in commercial products, the star-mesh topology shown in Figure 1. A star-mesh network consists of data sources (referred to as "endpoints") and relays denoted by R.

We restrict ourselves to the power management of the endpoints and relay nodes which are directly connected to these endpoints. The endpoints are equipped with sensors for designated event detection. When an endpoint detects an event, it will send a message to the relay. We assume

The authors' work is supported in part by the National Science Foundation under grant DMI-0330171, by AFOSR under grants FA9550-04-1-0133 and FA9550-04-1-0208, and by ARO under grant DAAD19-01-0610.

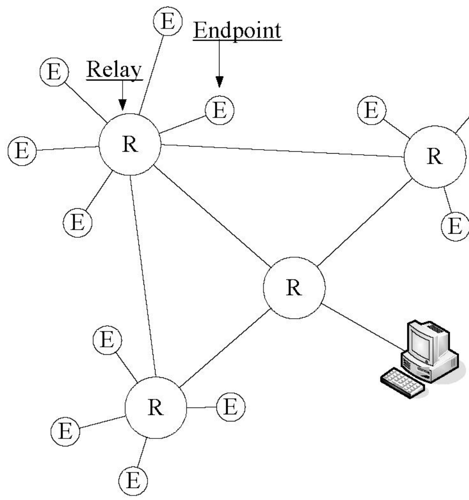

Fig. 1. A star-mesh network topology example

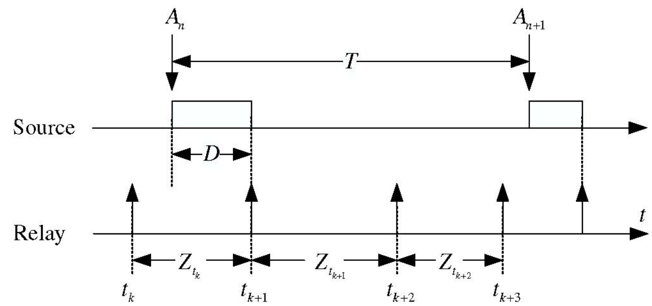

Fig. 2. Typical sample path

that events occur according to a renewal process whose interarrival time is random variable T and its associated cdf is $F(\tau)$. We intially assume that $F(\tau)$ is a priori known and later relax this assumption. We are interested in cases where arrivals are generally irregular and rare, so that the interarrival time is much larger than a typical sample period of the relay. We adopt the following mechanism for endpoint to relay communication:

1. If the relay node wakes up and samples the channel at time t, then it subsequently remains off for a period Z_t. At time $t + Z_t$, it turns on again for a very short time to check if the channel is busy. If it is, this means a message is to be transmitted by an endpoint so the relay should stay on until the message is received. If the channel is not busy, the relay node turns off again and repeats the process with a new sleep time. Note that the sleep time Z_t is time dependent. However, once determined at t it remains constant during the entire off period. Each sampling instant induces a fixed energy cost denoted by c.

2. When an endpoint detects an event, it starts the radio and sends a " preamble" until the relay is turned on and can receive the message to be transmitted. This implies that the message experiences a random delay, denoted by D, during which the preamble is persistently sent until the relay node responds. The source's energy cost is proportional to D. We define a constant r to denote the power consumed by an endpoint through this mechanism.

A typical sample path of this process is shown in Fig. 2 where A_n and A_{n+1} denote the epochs of the nth and $(n+1)$th event respectively. At time t_k, the relay samples the channel and finds it idle. It makes a decision to sleep for a time interval of length Z_{t_k}. Then, at time t_{k+1}, it samples the channel again. This time, the channel is busy since an arrival (A_n) has occured during $(t_k, t_{k+1}]$. Hence the message is transmitted. The transmission takes place in a very short time which will be ignored. Having received the message, the relay turns off again for an amount of time $Z_{t_{k+1}}$. Note that the actual transmission of the message also consumes energy. However, we do not consider this cost since it is uncontrollable, while in our analysis we aim to determine a suitable sleep time Z_t so as to reduce (i) the channel sampling cost, and (ii) the preamble cost, both of which depend on Z_t. The fundamental energy trade-off in this problem is the following. If the sleep time Z_t increases, the relay's energy consumption is reduced; on the other hand, the endpoints spend more time sending a preamble, thus expending more energy.

The controllable sleep time Z_t is fixed in such sleep time control schemes reported in the literature. However, knowledge of $F(\tau)$ can clearly be used to determine the likelihood of a message arrival over a certain period of time. Thus, it is natural to adjust the sampling density at the relay node according to this a priori information. This adjustment requires an estimation of future arrivals. Note that the delay D experienced by messages whose preamble is repeatedly transmitted at an endpoint depends on Z_t, so the first step is to identify the relationship between them.

Let t be a time instant when sampling has just taken place, and let the last event time be $A_n < t$. Based on the renewal process assumption, we rescale time for convenience so that $A_n = 0$ and t is the time elasped since the last event. When an event occurs, we can evaluate its expected delay conditioned on the fact that this arrival occurs in the interval $(t,\, t + Z_t]$ as follows:

$$E\left[D|t < T \leq t + Z_t\right] = t + Z_t - E\left[T|t < T \leq t + Z_t\right]$$
$$= t + Z_t - \frac{\int_t^{t+Z_t} \tau dF(\tau)}{F(t+Z_t) - F(t)} \quad (1)$$

It can be further observed that $E\left[D|t < T \leq t + Z_t\right]$ is continuous, non-decreasing in Z_t and unbounded. Therefore, given some $\bar{D} > 0$, where $\bar{D} = E\left[D|t < T \leq t + Z_t\right]$, there exists a solution $z > 0$ to (1) which allows us to make use of the arrival statistics. The solution z can be interpreted as the " longest" possible sleeping period under a given expected preamble time constraint. Therefore, if the next arrival is unlikely to happen over a short time horizon, z will be longer so as to adapt to the $\bar{D}$ constraint. This adaptation helps us reduce unnecessary channel sampling and hence energy cost. Lastly, if $E[D]$ is the unconditional expectation, then $E[D] = \bar{D}$. To see this, suppose the polling epochs form a sequence $\{z_n, n = 0, ..., N\}$ where $z_0 = 0, z_{n+1} = z_n + Z_{z_n}$ and N is the total number of polling epochs which can be infinite. The unconditional expected delay can be expressed as:

$$E[D] = \sum_i E[D|z_i < T \le z_{i+1}] P(z_i < T \le z_{i+1}) = \sum_i \bar{D} P(z_i < T \le z_{i+1}) = \bar{D} \quad (2)$$

which establishes the fact that under the control policy given by solving (1), the unconditional expected delay is also $\bar{D}$.

As we will see, this sleep time control approach lowers the total energy consumption of the system. However, it does not attempt to minimize it; it only dynamically selects sleep times that meet a specific delay through $\bar{D}$.

III. Uniform Distribution Example

To illustrate the benifit of considering the arrival statistics, we consider an example in which the message interarrival time distribution is uniform over $[a, b]$, where $0 \le a < b$. Here, using fixed sampling will waste energy polling the channel while it is not possible to have an arrival ($0 \le t \le a$). In this case, an analytical solution to (1) is obtainable. There are two cases to consider:

Case 1: $0 \le t \le a$. Equation (1) is rewritten as:

$$\bar{D} = t + Z_t - \frac{\int_a^{t+Z_t} \tau(b-a)^{-1} d\tau}{F(t+Z_t)} = \begin{cases} \frac{1}{2}(t + Z_t - a) & t + Z_t \le b \\ t + Z_t - \frac{1}{2}(a+b) & t + Z_t > b \end{cases} \quad (3)$$

Case 2: $a \le t \le b$. In this case, (1) is rewritten as:

$$\bar{D} = t + Z_t - \frac{\int_t^{t+Z_t} \tau(b-a)^{-1} d\tau}{F(t+Z_t) - (t-a)(b-a)^{-1}} = \begin{cases} \frac{1}{2}Z_t & t + Z_t \le b \\ Z_t - \frac{1}{2}(b-t) & t + Z_t > b \end{cases} \quad (4)$$

Letting $v = \max(t, a)$, the two cases combined give:

$$Z_t = \begin{cases} 2\bar{D} + v - t & \bar{D} \le \frac{1}{2}(b-v) \\ \bar{D} + \frac{1}{2}(b-t) + \frac{1}{2}(v-t) & \bar{D} > \frac{1}{2}(b-v) \end{cases} \quad (5)$$

In general, for an arbitrary distribution, the solution of (1) cannot be obtained in closed form. This is a critical issue in a limited-computation environment such as a WSN, and calls for approximation methods. On the other hand, in practice, a priori knowledge of the arrival distribution is unlikely, which leads to the necessicity of developing a " learning algorithm" for the relay to adapt to the arrival statistics based on observed data.

IV. Adaptive Algorithm for Arbitrary Distributions

To implement an adaptive algorithm for arbitrary distributions, we need to: (i) store statistical information, i.e., the interarrival time cdf; (ii) generate the control policy Z_t; (iii) learn the arrival statistics based on new arrival information. The last point suggests an iterative algorithm in which, with each message, timestamp information T and D is recorded and used to update the current approximation of the interarrival time cdf.

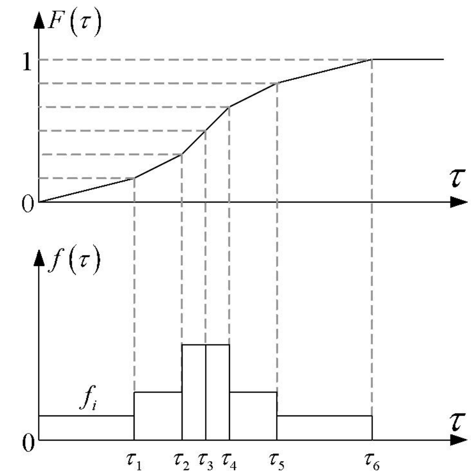

Fig. 3. Piecewise linear approximation of arbitrary distribution

A. Approximation of an arbitrary distribution

The approximation of an arbitrary distribution can be performed by sampling either the density or the distribution function. Since memory is scarce in WSNs, we require a finite length approximation. On the other hand, our approximation should also be able to cover an arbitrarily long support region of the underlying cdf. We choose to homogeneously sample the cdf along the y-axis over $[0, 1]$.

Figure 3 illustrates the approximation. In this figure, the arbitrary distribution $F(\tau)$ is approximated by $N = 6$ values, from τ_1 to τ_6. The values are selected such that: $F(\tau_i) = i/N, 1 \le i \le N$.

In other words, τ_i is the (i/N)th quantile of $F(\tau)$. If the underlying distribution has an infinite support region, $F(\tau_N)$ will have to be infinite. In this case, we choose a "large enough" value for τ_N, such as the $(1 - 0.1/N)$th quantile.

Points in (τ_i, τ_{i+1}) are linearly interpolated. Setting $\tau_0 = 0$, f_i is defined as the slope of $F(\tau)$ in the interval (τ_{i-1}, τ_i): $f_i = 1/[N(\tau_i - \tau_{i-1})]$, $i = 1, ..., N$, and hence is constant. Clearly, the underlying distribution is approximated by a finite mixture of uniform distributions. This allows us to derive a simple algorithm to solve (1).

B. Solving for Z_t

Based on the above approximation, we can calculate the integral in (1) by segments. For notational simplicity, let $u = t + Z_t$. To calculate the integral, we first need to determine in which interval (τ_{i-1}, τ_i), $i = 1, ..., N$, t and u lie. Define integers m, n such that: $1 \le m, n \le N$, $\tau_{m-1} < t \le \tau_m$, $\tau_{n-1} < u \le \tau_n$. Note that m, n may not exist satisfying these inequalities. For example, it is possible that $t > \tau_N$, or $u > \tau_N$. We will deal with these special cases separately.

Define: for $i = 1, ..., N-1$

$$A_m(t) = \int_t^{\tau_m} \tau dF(\tau) = \frac{f_m}{2}(\tau_m^2 - t^2)$$
$$B_i = \int_{\tau_{i-1}}^{\tau_i} \tau dF(\tau) = \frac{f_i}{2}(\tau_i^2 - \tau_{i-1}^2)$$
$$C_n(u) = \int_{\tau_{n-1}}^{u} \tau dF(\tau) = \frac{f_n}{2}(u^2 - \tau_{n-1}^2)$$

Then, it is clear that the integral $I(t,u) \triangleq \int_t^u \tau dF(\tau)$ can be rewritten in terms of $A_m(t)$, B_i, and $C_n(t,z)$:

$$I(t,u) = A_m(t) + \sum_{i=m+1}^{n-1} B_i + C_n(u) \tag{6}$$

Recall:

$$\bar{D} = u - \frac{I(t,u)}{F(u) - F(t)} \tag{7}$$

where $F(u)$ and $F(t)$ can also be calculated based on the linear interpolation:

$$F(t) = F(\tau_m) - f_m(\tau_m - t) = \frac{m}{N} - f_m(\tau_m - t)$$
$$F(u) = \frac{n}{N} - f_n(\tau_n - u)$$

We can see that $I(t,u)$ in (7) is quadratic in u and $F(u) - F(t)$ is linear in u. Noting that $F(u) - F(t) \neq 0$, (7) becomes a solvable quadratic equation in u. Before solving it, we need to identify n, which designates in which interval the solution u lies. Since $E[D|t < T < u]$ is non-decreasing in u, n must satisfy:

$$\tau_{n-1} - \frac{I(t,\tau_{n-1})}{F(\tau_{n-1}) - F(t)} < \bar{D} < \tau_n - \frac{I(t,\tau_n)}{F(\tau_n) - F(t)} \tag{8}$$

Therefore, n can be obtained by performing a search over all possible $m \leq n < N$. Once n is obtained, we are in position to solve for u. Define:

$$a_t = A_m(t) + \sum_{i=m+1}^{n-1} B_i - \frac{f_n}{2}\tau_{n-1}^2 \tag{9}$$
$$b_t = \frac{n}{N} + f_n\tau_n - \frac{m}{N} + f_m(\tau_m - t) \tag{10}$$

where a_t, b_t are calculable values depending on t. Thus, (7) can be rewritten as:

$$E[D|t < T < u] = u - \frac{\frac{1}{2}f_n u^2 + a_t}{f_n u + b_t} \tag{11}$$

As previously stated, this is solvable in u. Its solution is

$$Z_t = \frac{-(b_t - f_n\bar{D}) + \sqrt{(b_t - f_n\bar{D})^2 + 2f_n(a_t + b_t\bar{D})}}{f_n} - t \tag{12}$$

There are three special cases:

Case 1: $n = m$, that is, u and t both lie in the same interval $[\tau_{m-1}, \tau_m]$. Then,

$$\bar{D} = u - \frac{\frac{1}{2}f_m(u^2 - t^2)}{f_m(u - t)} = \frac{1}{2}(u - t) = \frac{1}{2}z$$

which gives:

$$Z_t = 2\bar{D} \tag{13}$$

Case 2: $u > \tau_N$, which is equivalent to:

$$\bar{D} > \tau_N - \frac{I(t,\tau_N)}{1 - F(t)} \tag{14}$$

This case implies that even letting $u = \tau_N$, which is the maximum possible interarrival time in the approximation, cannot satisfy the constraint set by $\bar{D}$. Since $I(t,u) = I(t,\tau_N)$ and $F(u) = 1$ for all $u \geq \tau_N$, the solution is

$$Z_t = \bar{D} + \frac{I(t,\tau_N)}{1 - F(t)} - t \tag{15}$$

Case 3: $t > \tau_N$. This case implies that the current interarrival time has exceeded the maximal possible interarrival time in the current approximation, which means the statistical knowledge about the interarrival time is insufficient. In this case, we take a conservative approach:

$$Z_t = \bar{D} \tag{16}$$

In conclusion, the approximation of an arbitrary distribution by a mixture of uniform distributions has introduced an attractive structure which has greatly eased the computational difficulty of solving (1). However, since a priori knowledge of the interarrival time distribution is usually unlikely, we need an algorithm to "learn" this distribution.

C. Updating the existing approximation

In what follows, we propose a mechanism to update the existing approximation based on new arrival information. The way in which the approximation is updated is essentially a quantile estimation problem. Our main concern, however, is to save memory as opposed to traditional quantile estimation methods that use sample quantiles. For example, [11] and [12] have proposed space-efficient recursive algorithms on quantile estimation based on stochastic approximation algorithms. Here, we propose an approach also based on the stochastic approximation method in [11], but modified to take advantage of the setting in our problem.

In [11], a stochastic approximation algorithm is used to solve an equation of the form $F(\tau) = \alpha$, where τ is the αth quantile of the unknown distribution function $F(\cdot)$. Suppose the observed values of random variable T are $\{T^k\}$. The estimator of $\tau = F^{-1}(\alpha)$ is given by:

$$\hat{\tau}^{k+1} = \hat{\tau}^k - \frac{d^k}{k+1}(\mathbf{1}\{T^{k+1} \leq \hat{\tau}^k\} - \alpha) \tag{17}$$

$$d^k = \min\left\{(\hat{\phi}^k)^{-1}, d^0 k^a\right\},\ 0 < a < \frac{1}{2} \tag{18}$$

where $\hat{\phi}^k$ is an estimate of the derivative of F at τ, given by:

$$\hat{\phi}^{k+1} = \frac{1}{k+1}(k\hat{\phi}^k + \frac{\mathbf{1}\{|T^{k+1} - \hat{\tau}^k| \leq h_{k+1}\}}{2h_{k+1}}) \tag{19}$$

where $\{h_k\}$ is a decreasing sequence satisfying $\sum_{k=1}^{\infty}(k^2 h_k)^{-1} < \infty$. It has been proved in [11] that estimator (17) converges to the αth quantile. In (19), $F'(\tau)$ is estimated by taking the average number of samples falling in a shrinking interval $\left[\hat{\tau}^k - h_{k+1}, \hat{\tau}^k + h_{k+1}\right]$. The issue with this estimation is that when samples are few, the convergence speed is slow since only local information is used. Also, one needs to carefully adjust $\{h_k\}$ so that it does not decrease too fast to collect enough samples falling in the interval. Since we need to estimate a set of quantiles and use them to approximate the whole function F, we can take advantage of the estimation of other quantiles in estimating $F'(\tau)$ around some particular quantile. For all $1 \le i \le N-1$,

$$\hat{\tau}_i^{k+1} = \hat{\tau}_i^k - \frac{d_i^k}{k+1}(\mathbf{1}\left\{T^{k+1} \le \hat{\tau}_i^k\right\} - \frac{i}{N}) \quad (20)$$

$$d_i^k = \min\left\{(\hat{\phi}_i^k)^{-1}, d^0 k^a\right\}, 0 < a < \frac{1}{2} \quad (21)$$

$$\hat{\phi}_i^{k+1} = \frac{2i}{N(\hat{\tau}_i^{k+1} - \hat{\tau}_i^k)} \quad (22)$$

and for $i = N$:

$$\hat{\tau}_N^{k+1} = \max\left\{\hat{\tau}_N^k, T^{k+1}\right\} \quad (23)$$

After we have obtained new $\hat{\tau}_i^{k+1}$, $i = 1, ..., N$, we can calculate f_i and go on to calculate Z_t with τ_i replaced by $\hat{\tau}_i^{k+1}$. We need to point out that (22) is *not* an unbiased estimator of $f(\tau_i)$, but a finite difference approximation. Simulation results (in the next section) show that the modified method is advantegeous in convergence speed.

V. Simulation Results

A. Under known distribution

We consider first an example assuming Gamma interarrival time cdf's of the same mean and different degrees of freedom. With this distribution, (1) has no analytical solution. Therefore, the approximation method proposed above is used. Since the Gamma distribution has infinte support, τ_N is selected as the (finite) 0.997th quantile.

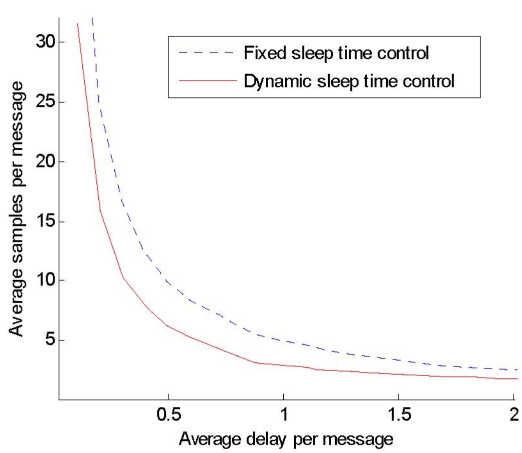

Fig. 4. Performance comparison between fixed sleep time control and dynamic sleep time control under Gamma $(20, 0.25)$ distribution.

Figure 4 illustrates the performance gain of using dynamic sleep time control. In the simulation, we fixed the interarrival distribution to be Gamma with parameter $(20, 0.25)$ and used the same random seed for both controls. Then, different Z or $\bar{D}$ corresponding to the two policies are chosen. The figure is plotted as the average number of samples, $\bar{K}$, needed to deliver one message as a function of the average delay $\bar{D}$ per message. We chose to measure these quantities, as opposed to directly measuring energy, because $\bar{K}$ and $\bar{D}$ are more fundamental. First, $\bar{K}$ and $\bar{D}$ are directly linked to energy consumption through the coefficients c and r defined in Section II, i.e., $P_{source} = r\bar{D}/E[T]$, $P_{relay} = c\bar{K}/E[T]$.

Second, by looking at Fig. 4, one can easily see the advantage of using dynamic sleep time control: the solid curve is always below the dotted curve. Given any $\bar{D}$ on the x-axis, using dynamic sleep time control always requires fewer pollings per message than the fixed sleep time control which yields the same average delay.

Clearly, the ultimate goal is to optimize the total energy cost. The following table compares the optimal energy cost under the same Gamma distribution with parameters (a, b), same normalized energy coefficients, and two sleep time control policies. In the table, the "Fixed" and "Dynamic" column denote the minimum energy consumption (obtained by exhaustive search) under the corresponding control policy.

Distribution	r/c	Fixed	Dynamic	Save%
$(20, 0.25)$	2	0.89	0.79	11.24
$(20, 0.25)$	10	2.00	1.89	5.50
$(20, 0.25)$	50	4.40	4.39	0.23
$(10, 0.5)$	10	1.97	1.88	4.57

The results show that the power saving by using dynamic sleep time control increases as the ratio r/c decreases. This justifies the use of statistical information: if it is expensive to poll the channel, the polling time should be carefully chosen based on all the information. On the other hand, if polling is cheap, it is not worth the computation to use dynamic sleep time control. The last row shows that the saving decreases as the degree of freedom in the Gamma distribution decreases, which is expected since more randomness is introduced.

B. Adaptive algorithm

In the absence of complete knowledge of the interarrival time distribution, we need to use the updating scheme proposed in the last section to estimate it. Figs. 5 and 6 show simulation results for a system with a Gamma distribution with parameters $(20, 0.25)$, which are *unknown* to the controller. Figure 5 shows that compared to [11], the method we have proposed has faster convergence speed. This is important since the quality of control increases faster as the root-mean-square error (RMSE) decreases. In Fig. 6, the total power (source and relay) is measured by taking a moving average of the instantaneous power. The process begins with a uniform distribution approximation $U[0, 10]$. Since the approximation error is large, the total power during this period is high. As time evolves and the interarrival times are

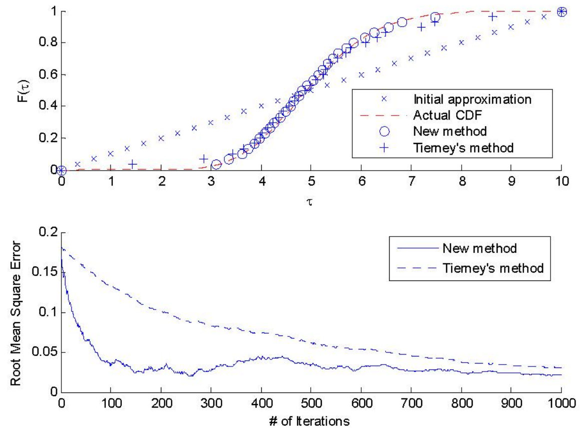

Fig. 5. Comparison of the convergence speed between the new method and Tierney's method.

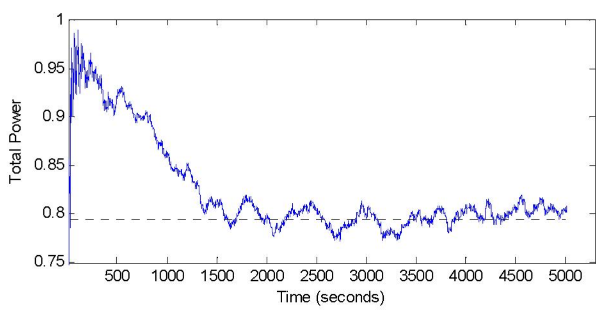

Fig. 6. Dynamic sleep time control with interarrival time distribution estimation

collected, the approximation is updated and better controls are generated, which subsequently reduces the power.

VI. Conclusion and Future Work

In this paper, we have presented a dynamic sleep time control method to reduce channel sampling cost and source preamble cost. Dynamic sleep time control transforms the original fixed sleep time control problem into a new one which fixes the "expected delay" as a parameter and finds the corresponding dynamic sleep time. The benefit of the new method comes from exploiting event time distributional information. Simulation results show that by using this information, the performance, in terms of energy consumption, is superior to fixed sleep time control. Since in WSN, it takes much more energy to transmit/receive than CPU calculation, it pays off to perform some optimization routines and make careful decision on using the radio wisely. In addition, we have addressed two issues, (i) the computational difficulty in solving equation (1), and (ii) the absence of a priori statistical information, by proposing an method to approximate an arbitrary distribution and a stochastic approximation algorithm which updates the approximation with new arrival information. As the iterative algorithm involves only (20)–(23) and solving for Z_t involves only (9)–(16), the calculation is indeed lightly loaded.

We must point out that the dynamic sleep time control method presented in this paper is not optimal. In fact, the sleep time obtained by solving (1) is a conservative one since it guarantees the expected delay conditioning on the arrival occurs during the sleep time, which is a strict constraint. Ongoing work aims at relaxing this constraint while still maintaining a fixed unconditional expected delay, hence possibly archieving better energy savings. Note that currently the key parameter $\bar{D}$ is assumed given. To find an optimal sleep time control policy such that the power consumption is minimized, a mechanism to find the optimal parameter $\bar{D}^*$ is also being sought for.

References

[1] S. Megerian and M. Potkonjak, *Wireless Sensor Networks*, ser. Wiley Encyclopedia of Telecommunications, December 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] B. Hohlt, "The design and evaluation of network power scheduling for wireless sensor networks," Ph.D. dissertation, University of California, Berkeley, 2005.

[4] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "The design of an acquisitional query processor for sensor networks," in *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data.* New York, NY, USA: ACM Press, 2003, pp. 491–502.

[5] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *The First ACM Conference on Embedded Networked Sensor Systems (Sensys03)*, November 2003.

[6] A. El-Hoiydi and J.-D. Decotignie, "WiseMAC: An ultra low power mac protocol for multi-hop wireless sensor networks," *Lecture Notes in Computer Science*, vol. 3121, pp. 18–31, Jan 2004, springer-Verlag GmbH.

[7] B. Mangione-Smith, "Low power communications protocols: paging and beyond," in *IEEE Symposium on Low Power Electronics*, October 1995, pp. 8–11.

[8] A. El-Hoiydi, "Spatial TDMA and CSMA with preamble sampling for low power ad hoc wireless sensor networks," in *Proceedings of IEEE Seventh International Symposium on Computers and Communications*, July 2002, pp. 685–692.

[9] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems.* New York, NY, USA: ACM Press, 2004, pp. 95–107.

[10] M. Sichitiu, "Cross-layer scheduling for power efficiency in wireless sensor networks," in *Proceedings of IEEE INFOCOM 2004*, March 2004, pp. 1740–1750.

[11] L. Tierney, "A space-efficient recursive procedure for estimating a quantile of an unknown distribution," *SIAM Journal of Sci. Stat. Comput.*, vol. 4, no. 4, pp. 706–711, December 1983.

[12] F. Chen, D. Lambert, and J. C. Pinheiro, "Incremental quantile estimation for massive tracking," in *Knowledge Discovery and Data Mining*, 2000, pp. 516–522.